COMPUTER BRAILLE CODE

Revised 2000

Developed Under the Sponsorship of the
BRAILLE AUTHORITY OF NORTH AMERICA

2000

BANA MEMBERS

American Council of the Blind (ACB)
American Foundation for the Blind (AFB)
American Printing House for the Blind (APH)
Associated Services for the Blind (ASB)
Association for Education and Rehabilitation of the Blind and Visually Impaired (AER)
California Transcribers and Educators of the Visually Handicapped (CTEVH)
The Canadian National Institute for the Blind (CNIB)
The Clovernook Center: Opportunities for the Blind (CCOB)
National Braille Association (NBA)
National Braille Press (NBP)
National Federation of the Blind (NFB)
National Library Service for the Blind and Physically Handicapped, Library of Congress (NLS/BPH)

Royal New Zealand Foundation for the Blind (RNZFB), Associate Member

Copyright © 1987, 2000 by the Braille Authority of North America.
This material may be duplicated but not altered.

Printed by the American Printing House for the Blind. Copies may be purchased from:

American Printing House for the Blind
P.O. Box 6085
Louisville, KY 40206-0085
502-895-2405
FAX: 502-899-2274
E-Mail: info@aph.org
Catalog Number: 7-24410-00

CONTENTS

FOREWORD vii
FOREWORD TO THE 2000 EDITION viii
CHANGES AND CLARIFICATIONS TO THE COMPUTER BRAILLE CODE IN 2000 ix

I.0 GUIDELINES FOR IDENTIFYING COMPUTER NOTATION 1
1.1 A Note 1
Example 1.1.1 Note in Text 1
Example 1.1.2 Dot Matrix Print 2
1.2 Program Lines 2
Example 1.2.1 Short Program Lines Identified by Numbers 2
Example 1.2.2 Systematic Indentation of Program Lines 3
1.3 Comments 3
Example 1.3.1 Explanatory Comments 4
Example 1.3.2 Explanatory Comments 4
1.4 Words 4
Example 1.4.1 Word Enclosed within Angle Brackets 4
1.5 Horizontal Signs 5
Example 1.5.1 Horizontal Sign of Comparison 5
Example 1.5.2 Horizontal Sign of Comparison 5
1.6 Superimposed Symbols 5
Example 1.6.1 Superimposed Symbol 5
1.7 Words and/or Symbols 6
Example 1.7.1 Words Out of Context 6
Example 1.7.2 Symbol Not Represented in Primary Code 6

2.0 COMPUTER BRAILLE CODE SYMBOLS AND USAGE 7
2.1 The Basis of the Computer Braille Code 7
2.2 Use of Braille Symbols 7
2.3 The Computer Braille Code—A Separate Code 7
Example 2.3.1 Technical-Appearing Text (English Braille) 8
Example 2.3.2 Text with Embedded Computer Notation 8
Example 2.3.3 Text with Embedded Web Site Address 8
Example 2.3.4 Computer Program 9
Example 2.3.5 Nonuse of to, into, and by Preceding Computer Braille Code Indicator 9
Example 2.3.6 Nonuse of to, into, and by Preceding Computer Braille Code Indicator 9

3.0 FORMAT 11
3.1 Computer Braille Code Transcriptions 11
3.2 Transcriber's Note 11
3.3 Symbols 11
3.4 Blank Lines 11
Example 3.4.1 Computer Notation Set Off from Body of Text 12
Example 3.4.2 Computer Notation (Web Site Address) Set Off From Body of Text 13
3.5 Entire Computer Program 13
Example 3.5.1 Program with Identifying Label 14
3.6 Computer Notation within Text 17
Example 3.6.1 Embedded Computer Notation 17
Example 3.6.2 Embedded Computer Notation with Runover 17
Example 3.6.3 Embedded Computer Notation (Web Site) with Runover 18
Example 3.6.4 Embedded Computer Notation on New Line 18

4.0 UPPER AND LOWER CASE DESIGNATION 19
4.1 Computer Notation/Lower Case 19
Example 4.1.1 Single BASIC Program Transcribed without Case Indicators 19
4.2 Shift Indicator 19
4.3 Caps Lock Indicator 20
Example 4.3.1 Consecutive and Single Upper Case Letters 20
Example 4.3.2 Case Designation in E-mail Address 20
4.4 Caps Release Indicator 20
Example 4.4.1 Caps Lock and Caps Release Indicators 20

5.0 REPRESENTATION OF CONTROL CHARACTERS 21
5.1 Control Characters 21
Example 5.1.1 "Carriage Return" Representation 21
Example 5.1.2 "Carriage Return" Representation 21

6.0 CONTINUATION INDICATOR 23
6.1 Continuation Indicator 23
Example 6.1.1 Continuation Indicator 23
6.2 Runovers of Computer Notation 23
6.3 Continuation Indicator 24
Example 6.3.1 Continuation Indicator within Embedded Computer Notation 24
Example 6.3.2 Continuation Indicator within Embedded Web Site Address 24

7.0 INDENTED MATERIAL 25
7.1 Levels of Indentation 25
7.2 Runovers 25
Example 7.2.1 Indented Material 25
Example 7.2.2 Indented Material 26

8.0 ISOLATED LOWER-CELL SIGNS 27
8.1 Isolated Lower-Cell Signs 27
Example 8.1.1 Isolated Lower-Cell Signs 27

9.0 COUNTABLE SPACES 29
9.1 Countable Spaces Indicator 29
Example 9.1.1 Countable Spaces Indicator 29
9.2 Countable Spaces with Runovers 30
Example 9.2.1 Countable Spaces with Continuation Indicator 30
Example 9.2.2 Countable Spaces Beginning on Runover Line 31

10.0 EMPHASIS INDICATORS 33
10.1 Emphasis Indicators 33
Example 10.1.1 Emphasis Indicators 33
Example 10.1.2 Emphasis Indicators 33
10.2 Type Style 33
Example 10.2.1 Emphasis Indicators with More Than One Type Style 34
Example 10.2.2 Emphasis Indicators with More Than One Type Style 35

11.0 COMPUTER BRAILLE CODE INDICATORS 37
11.1 Computer Braille Code Indicators 37
11.2 Caps Lock Indicator 37
Example 11.2.1 Computer Braille Code Indicators 37
Example 11.2.2 Computer Braille Code Indicators 37
Example 11.2.3 Avoiding Frequent Switches Between Computer Braille Code and Primary Code 38
Example 11.2.4 Use of Computer Braille Code Indicators to Clarify Ambiguities 38

12.0 NEMETH CODE INDICATORS 39
12.1 Nemeth Code 39
Example 12.1.1 Nemeth Code Indicators 40
Example 12.1.2 Nemeth Code Indicators with Horizontal Fraction Line 41
12.2 Nemeth Code Indicators 41
Example 12.2.1 Nemeth Code Indicators within Computer Braille Code 41

13.0 SHAPE INDICATORS 43
13.1 Shape Indicators 43
Example 13.1.1 Shape Indicators 43
Example 13.1.2 Shape Indicators 43
13.2 Computer Keyboard 43
Example 13.2.1 Depicted Key 44
13.3 Transcribers of Nemeth Code 44
Example 13.3.1 Use of Nemeth Code within Shape Indicators 44
13.4 A Transcriber's Note 44

14.0 TRANSCRIBER'S OPTION SYMBOLS 45
14.1 Transcriber's Option Symbol 45
Example 14.1.1 Transcriber's Option Symbol Representing Non-ASCII Character (b) 45
14.2 Transcriber's Substitution 45
Example 14.2.1 Transcriber's Substitution (ASCII Tilde Becomes EBCDIC "Not" Sign) 45

15.0 HALF-LINE SHIFTS 47
15.1 Half-Line Shifts Up 47
15.2 Half-Line Shifts Down 47
Example 15.2.1 Half-Line Shifts 48
Example 15.2.2 Half-Line Shifts Terminated by A Space 48
15.3 Reference Indicators 49
Example 15.3.1 Reference Indicator (English Braille) 49
Example 15.3.2 Reference Indicator (Nemeth Code) 49

16.0 TERMINATION INDICATOR 51
16.1 Termination Indicator 51
Example 16.1.1 Termination Indicator 51

COMPUTER BRAILLE CODE SYMBOLS: ASCII ORDER 53
Additional Computer Braille Code Symbols 60

COMPUTER BRAILLE CODE SYMBOLS: BRAILLE ORDER 63

INDEX 73

FOREWORD

As our society's dependence on computing technology grows, computers become increasingly important to the blind and visually impaired. Those who read braille in order to learn about computers and to communicate with them have an urgent need for a braille computer code that is unambiguous, and that makes possible the faithful transcription into braille of computer-related text in print. In 1972, BANA recognized this need by adopting The Provisional Braille Code for Computer Notation. It was a preliminary code that left many problems unsolved, but its adoption allowed braille readers and braille transcribers to accumulate a body of experience on which to base a more efficient code. The code presented and explicated in this volume has been made possible by that experience.

The Computer Braille Code utilizes the sixty-four combinations of dots that can be formed in the braille cell, and assigns to these dot patterns the same meanings assigned to them by North American manufacturers of computer braille equipment. However, an effective braille code must be capable of representing the 128 characters in the ASCII code, and must provide a few additional symbols that transcribers need in order to achieve faithful braille transcriptions. To meet these requirements, two meanings must be assigned to many of the sixty-four dot patterns, and a way of indicating the meaning that is intended must be provided. In the Computer Braille Code, the differentiation between the two meanings assigned to a dot pattern is accomplished by means of a prefix consisting of dots 456.

The Computer Braille Code Committee has studied a large and diverse sample of current computer materials, and has worked diligently to make the Computer Braille Code a realistic code, capable of unambiguous representation of current computer notation, but flexible enough to respond to changing and expanding needs. Feedback from braille readers, transcribers, and publishers has been solicited, and is reflected in this edition.

BANA and this Committee recognize that the code does not address important issues, such as the transcription of flow charts and graphics. However, we are pleased to publish it in its present form, and trust that a future revision will deal with such issues. We welcome suggestions and comments from the field.

BANA wishes to thank Dr. T. V. Cranmer (Chairman of the Computer Braille Code Committee), and Dr. Emerson Foulke, Mrs. Priscilla Harris, Miss Donna Pastore, Dr. Sandra Ruconich, Dr. Lawrence Scadden, and Mr. Joseph Sullivan (committee members), for their work.

FOREWORD TO THE 2000 EDITION

Since the 1987 publication of the Computer Braille Code, the use of the internet has become commonplace. This edition contains some new, internet-related examples which did not exist in 1987.

The numbering of rules and examples has changed to reflect the BANA document developed in 1998, SPECIFICATIONS FOR THE PRODUCTION OF BRAILLE AND PRINT PUBLICATIONS FOR THE BRAILLE AUTHORITY OF NORTH AMERICA. Each section stands alone, making future additions easier.

Section 3.6, the rule that deals with the placement of embedded computer notation, has been clarified to address the issue of where to begin a passage of embedded computer notation which requires more than one braille line.

The new examples and the clarified rule are listed on page ix, and an index has been added to the code for easier reference.

BANA COMPUTER TECHNICAL COMMITTEE

Eileen Curran (Consultant)
Edward Godfrey
Melissa Hirshson (Consultant)
John Jackson
Dr. Sandra Ruconich
Joseph Sullivan
Betty Teachman (Chair)

CHANGES AND CLARIFICATIONS TO THE COMPUTER BRAILLE CODE IN 2000

Example 1.3.1d changed
Example 1.7.2 new

Example 2.3.3 new
Example 2.3.6 new

Example 3.4.2 new
Example 3.5.1 new
Section 3.6 clarified rule
Example 3.6.3 new
Example 3.6.4 changed

Example 4.3.2 new

Example 6.3.2 new

Example 10.2.2 new

Example 11.2.2 new

Example 13.1.2 new

1.0 GUIDELINES FOR IDENTIFYING COMPUTER NOTATION

The Computer Braille Code is entered when precise representation is essential. Thus, material such as programs, program lines, computer commands, and filenames should be transcribed in Computer Braille Code. The following guidelines may be helpful in the identification of computer notation. Because languages and situations the transcriber may encounter will doubtless vary widely, these guidelines are offered merely as suggestions and not as all-inclusive, binding rules.

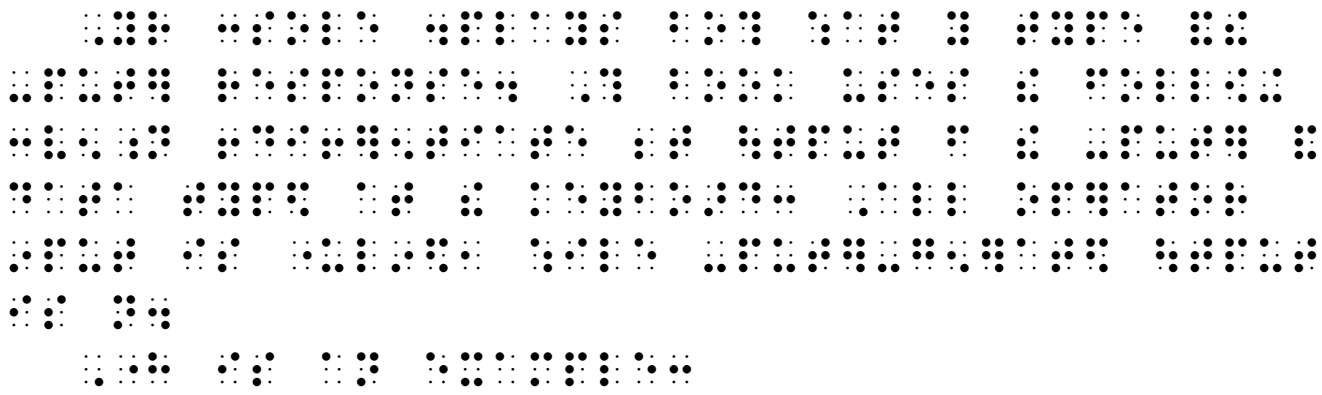
1.1 A note at the beginning of a work may describe how computer notation is represented. Even if no such note exists, stylistic practices (e.g., type styles different from surrounding text—dot matrix print, for instance) may indicate the presence of computer notation.

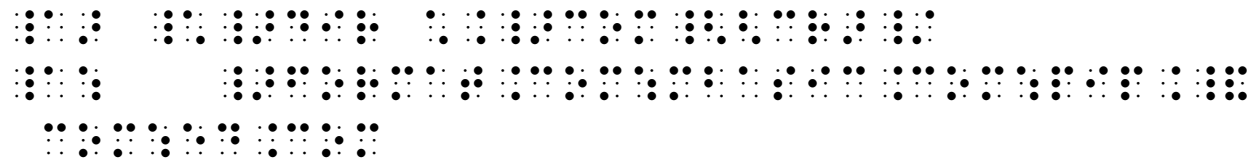
Example 1.1.1: Note in Text

Your console displays both what you type and the computer response. This book uses the following convention to differentiate between output from the computer and data typed at the keyboard: All operator input is underlined, while computer-generated output is not.

Here is an example:

A> DIR *.COM<cr>
A: FORMAT.COM:MBASIC.COM:PIP.COM:ED.COM





Example 1.1.2: Dot Matrix Print

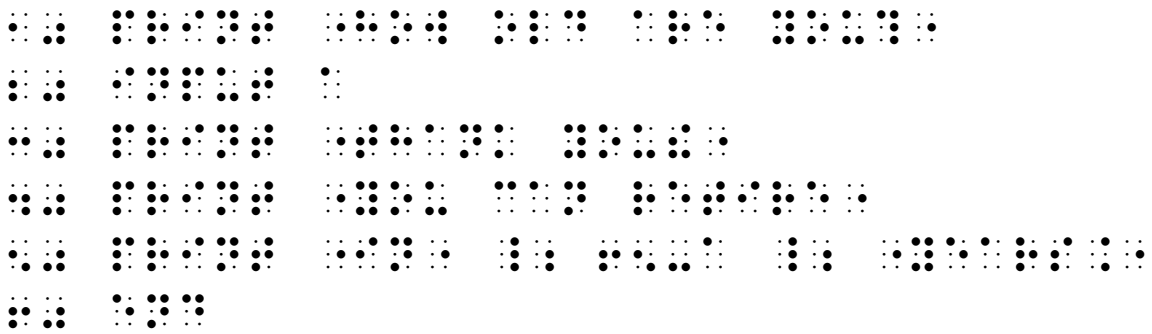
```
DOS READY
```



1.2 Program lines are generally relatively short and sometimes begin with numbers. Often these lines are indented in a systematic way which distinguishes them from mathematics or regular text.

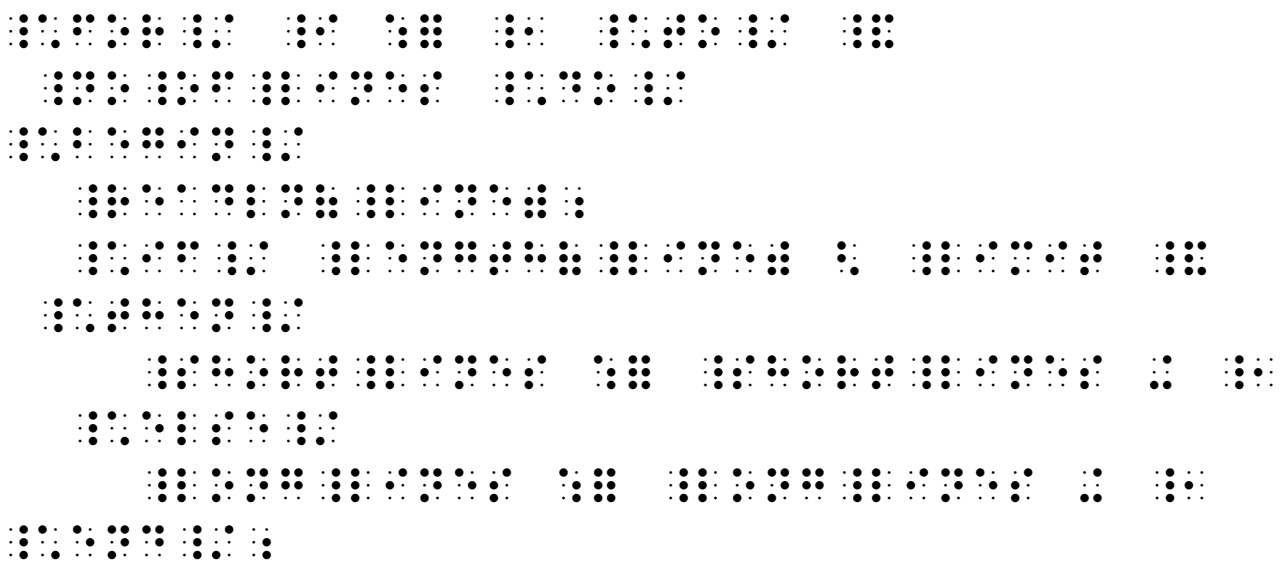
Example 1.2.1: Short Program Lines Identified by Numbers

```
10 PRINT "HOW OLD ARE YOU?"
20 INPUT A
30 PRINT "THANK YOU!"
40 PRINT "YOU CAN RETIRE"
50 PRINT "IN" ; 65-A ; "YEARS."
60 END
```



Example 1.2.2: Systematic Indentation of Program Lines

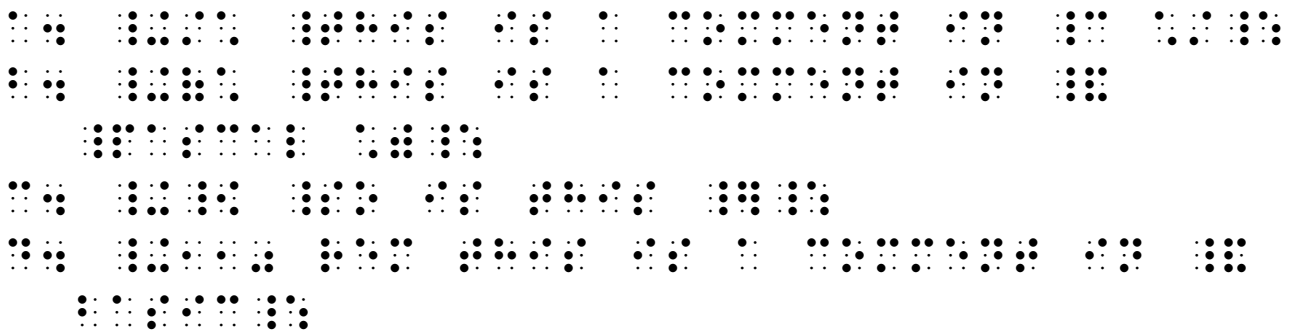
```
for I := 1 to NoOfLines do
begin
   Readln(Line);
   if Length(Line) < Limit then
      ShortLines := ShortLines + 1
  else
      LongLines := LongLines + 1
end;
```



1.3 Comments sometimes explain what is happening in a program. These comments are frequently distinguished from surrounding material by characteristic symbols or words [e.g., /* ... */, (* ... *), REM, OUTPUT, etc.].

Example 1.3.1: Explanatory Comments (5-00)

a. /* This is a comment in C */
b. (* This is a comment in Pascal *)
c. { So is this }
d. 110 REM THIS IS A COMMENT IN BASIC



Note: These program lines are treated as embedded in the examples because of the identifying letters. In an actual program listing they would be in displayed notation, and the indicators would not be used.

Example 1.3.2: Explanatory Comments

{ Longest possible representation }



1.4 Words may be enclosed within quotation marks or angle brackets.

Example 1.4.1: Word Enclosed within Angle Brackets

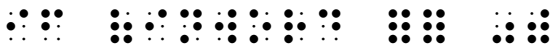
<FILE>



1.5 Horizontal signs of comparison (e.g., <=, ==, :=) may be present.

Example 1.5.1: Horizontal Sign of Comparison

if (inword == 0)



Example 1.5.2: Horizontal Sign of Comparison

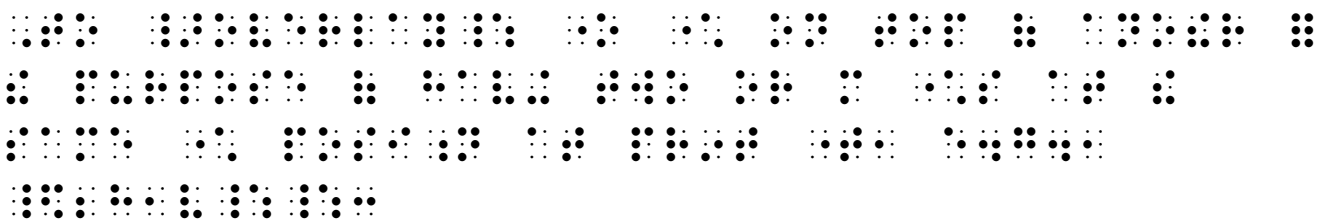
(ev ~= evNull)=>



1.6 Superimposed symbols (e.g., a circle with a slash inside) may appear.

Example 1.6.1: Superimposed Symbol

To OVERLAY one character on top of another for the purpose of having two or more characters at the same character position at print time, e.g., ≠:



Note: The shape indicators enclose 2h1v to describe the shape consisting of two horizontal lines and one vertical line.

1.7 Words and/or symbols may not make sense in the context of the primary code being used.

Example 1.7.1: Words Out of Context

10 LPRINT "HELLO"



Example 1.7.2: Symbol Not Represented in Primary Code (5-00)

bruce_r._fortnum@magic.ca



Note: This E-mail address contains underscore characters (456, 456) which are not represented in English Braille.

2.0 COMPUTER BRAILLE CODE SYMBOLS AND USAGE

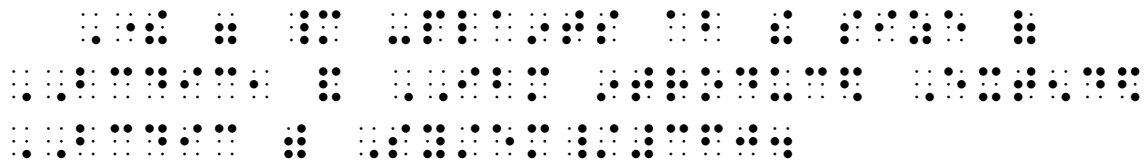
2.1 The Computer Braille Code is based on ASCII (American Standard Code for Information Interchange), a code used by virtually all computer-related braille devices distributed in North America. Printable symbols from other codes (e.g., EBCDIC and BAUDOT) can also be represented using Computer Braille Code symbology. The Computer Braille Code makes communication with braille devices possible while simultaneously providing for the precise transcription of computer-related materials in braille. A table of Computer Braille Code symbols appears at the end of this code.

2.2 Although in certain instances many braille devices use one braille symbol to represent two different print characters (e.g., upper case A and lower case a are both brailled as dot 1), the Computer Braille Code avoids this ambiguity. Standard alphabetic characters (a-z) are used for lower case letters. The same symbols, preceded by dots 456, are used to represent upper case letters. Braille symbols used in this dual manner include single upper case (A) and lower case (a) letters, at sign (@) and grave accent (`), up arrow/circumflex (^) and tilde (~), left bracket ([) and left brace ({), right bracket (]) and right brace (}), and backslash (\) and vertical bar (|). The underscore symbol (_) has a separate identity.

2.3 The Computer Braille Code is a separate, secondary code, which may be used in conjunction with the primary code in which the text is transcribed—whether that code be English Braille, Braille Formats, Nemeth, or Braille Music. Material such as numbers, abbreviations, acronyms, contractions, and punctuation marks should be transcribed in this primary code unless there is good reason to enter Computer Braille Code. Thus, technical-appearing abbreviations and acronyms such as "IBM" and "ASCII" occurring within an English Braille transcription should be transcribed in English Braille. Computer Braille Code is entered when precise representation is essential; for this reason, Computer Braille Code material is transcribed character for character (i.e., no contractions are used), and the contractions to, into, and by may not be used before any Computer Braille Code indicator. Computer Braille Code numbers are transcribed in the lower two-thirds of the cell and are not preceded by number signs. Programs and other computer notation set off from the body of the text and computer input in which every letter must be typed precisely in the manner shown are examples of material which should be transcribed in Computer Braille Code. When in doubt, Computer Braille Code should be used. Likewise, the transcriber must also judge or seek advice as to the significance of vertical and horizontal spacing within the text. When in doubt, assume spacing is significant.

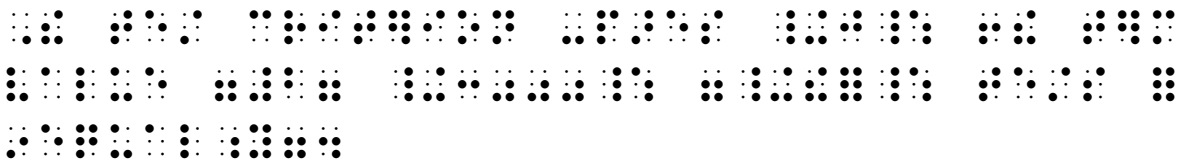
Example 2.3.1: Technical-Appearing Text (English Braille)

There were many complaints about the size of BCDIC, and IBM introduced Extended BCDIC with System/360.



Example 2.3.2: Text with Embedded Computer Notation (English Braille and Computer Braille Codes)

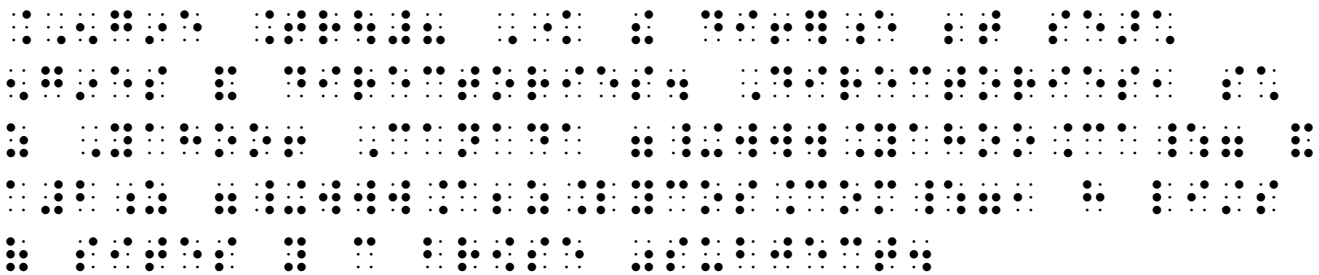
The test criterion compares j to the term value (2) 3000 (!= tests for inequality).



Note: The (2) refers to a second item of previously discussed material.

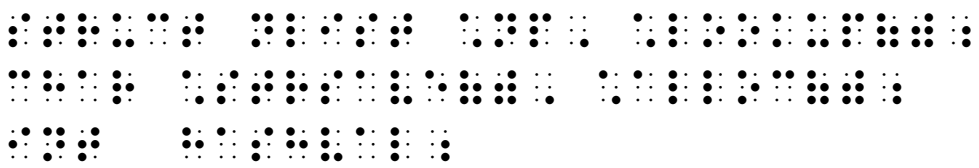
Example 2.3.3: Text with Embedded Web Site Address (English Braille and Computer Braille Codes) (5-00)

Engine trouble? Know the difference between search engines and directories. Directories, such as Yahoo! Canada (www.yahoo.ca) and a2z (www.a2z.lycos.com), have lists of sites you can browse by subject.



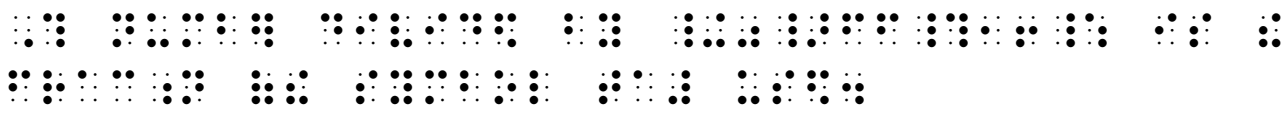
Example 2.3.4: Computer Program (Computer Braille Code)

```
struct nlist *np, *lookup();
char *strsave(), *alloc();
int   hashval;
```



Example 2.3.5: Nonuse of to, into, and by Preceding Computer Braille Code Indicator

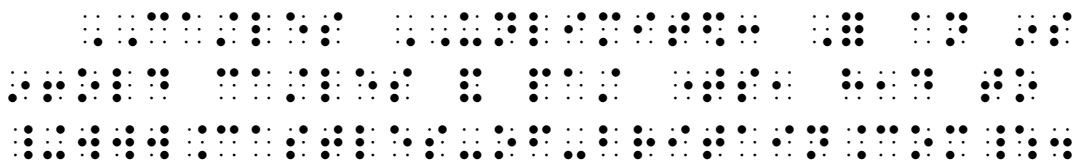
This number divided by $0FF_{16}$ is the fraction of the symbol table used.



Note: This example is in hexadecimal notation. In this notation, the numbers are the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *and* the letters A, B, C, D, E, F. Therefore, the example shows the number 0FF not the word OFF.

Example 2.3.6: Nonuse of to, into, and by Preceding Computer Braille Code Indicator (5-00)

CASTLES UNLIMITED: For an insight into old castles and past times, head to www.castles-of-britain.com.



3.0 FORMAT

3.1 Computer Braille Code transcriptions should use the longest possible braille line allowed by production and duplication considerations. Current line length is typically 40 or 41 cells.

3.2 A transcriber's note at the beginning of each volume of a book should state the year of adoption of the Computer Braille Code, as well as the primary code in which the book is transcribed. If a book requires more than two codes, all codes employed should be listed. (For other information required on the Transcriber's Notes Page, see also 4.1, 10.2, 13.4, 14.1, and 14.2).

3.3 Symbols which the transcriber has devised or those assigned special usage should be explained or drawn on a "Special Symbols" page. These symbols should be transcribed in accordance with the rules of the Braille Formats Code. Neither standard Computer Braille Code symbols nor symbols of other adopted codes which are used in accordance with the rules of the applicable code(s) should be included on the "Special Symbols" page, except as may be required by applicable codes.

3.4 A blank line should precede and follow programs and computer notation set off from the body of the text. Transcription of such material should begin in cell 1 of the appropriate line, with runovers beginning in cell 2 of the following line(s). (See Sections 6.0, 7.0, and 9.0 for more information about runovers.)

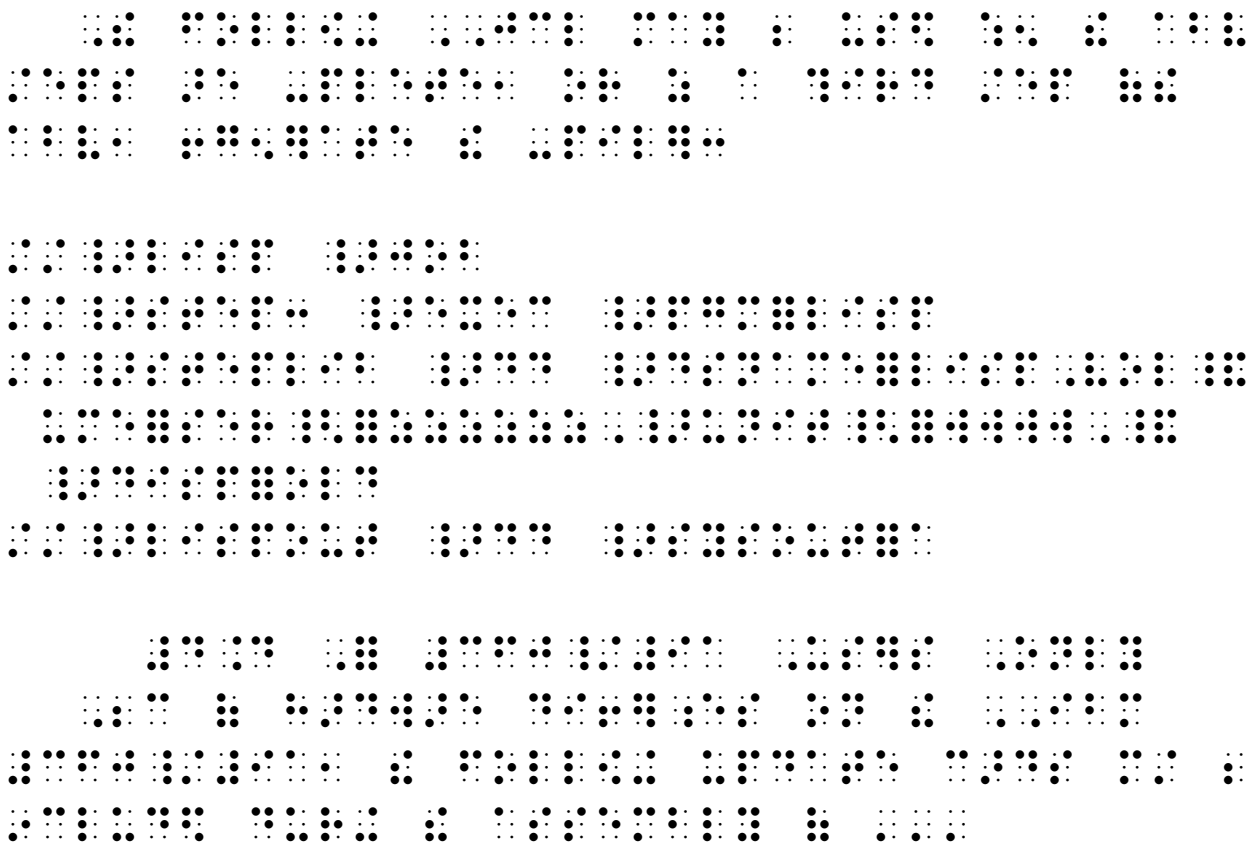
Example 3.4.1: Computer Notation Set Off from Body of Text

The following JCL may be used when the above steps are complete, or as a third step of the above, to generate the compiler:

```
//LISP JOB
//STEP3 EXEC PGM=LISP
//STEPLIB DD DSNAME=LISP,VOLUME=SER=zzzzzz,UNIT=wwww,DISP=OLD
//LISPOUT DD SYSOUT=A
```

4.4 For 360/91 Users Only

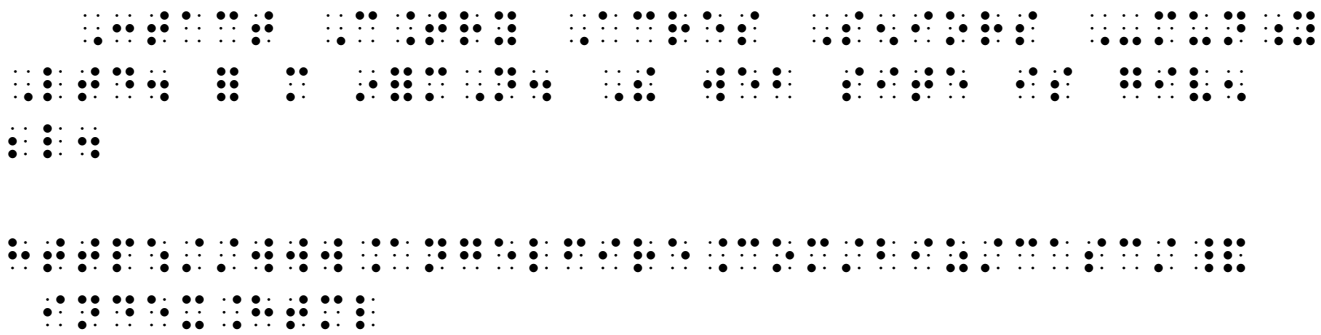
Because of hardware differences on the IBM 360/91, the following update cards must be included during the assembly of ...



Example 3.4.2: Computer Notation (Web Site Address) Set Off from Body of Text (5-00)

Contact Country Acres Seniors Community Ltd. for more information. The web site is given below.

http://www.angelfire.com/biz/casc/index.html

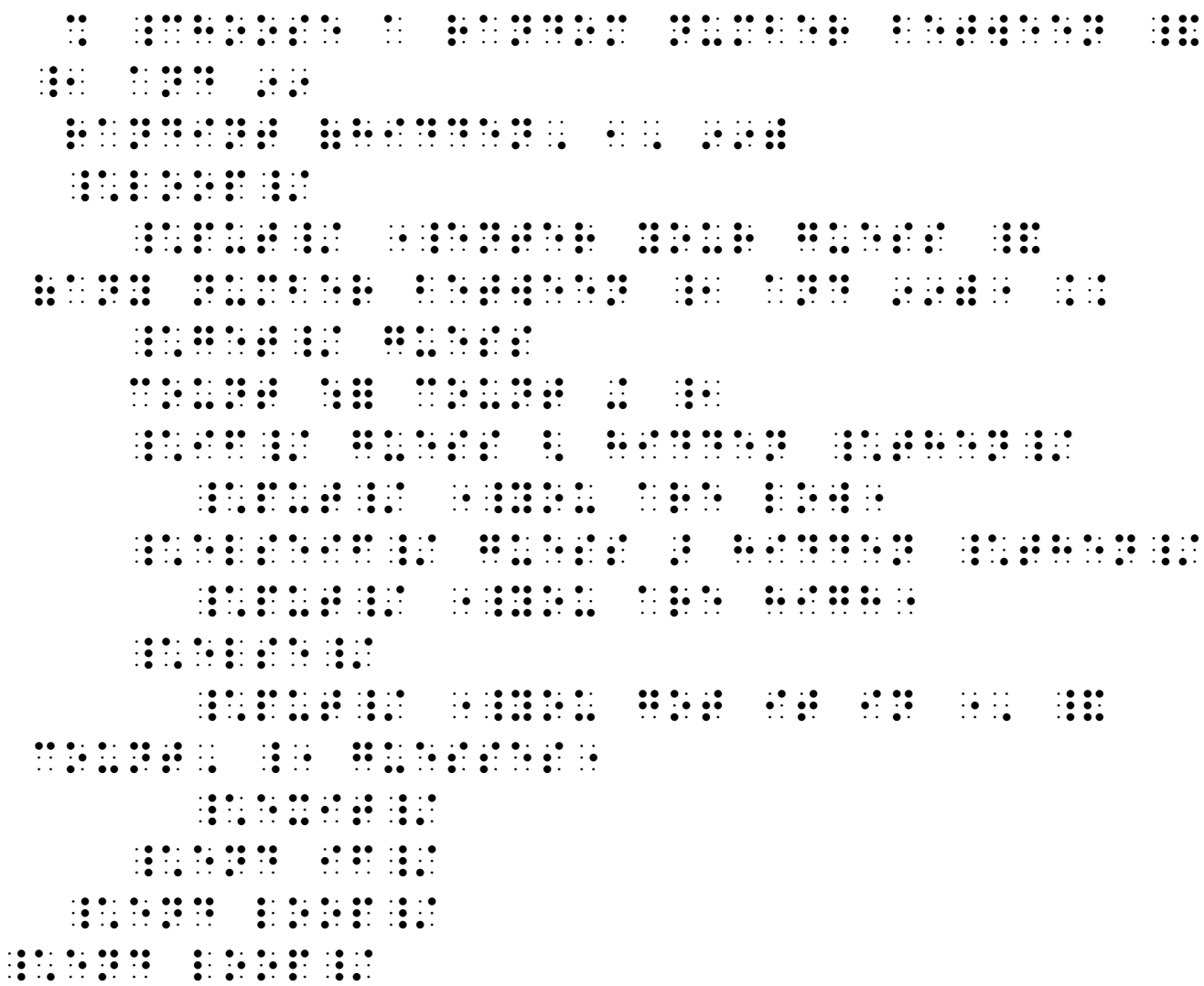


3.5 Whenever possible, an entire computer program should be transcribed on a single braille page. When a program is too long to fit on a single page, any identifying label should be centered on the last line (usually line 25) of each page to which it applies. A program line cannot be divided between pages. Text following a program requiring more than one braille page should begin on a new page, with the transition to the new page replacing the blank line which ordinarily follows a program.

Example 3.5.1: Program with Identifying Label (5-00)

Program 3-5
"Guess the Number" Program

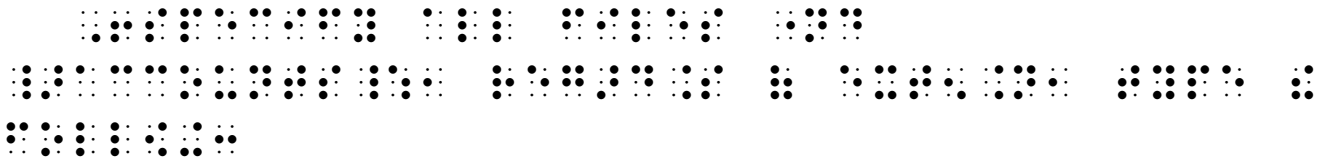
```
% The "Guess the Number" program
% Chooses a random number between 1 and 99
% and allows you to guess it
randomize
var hidden, guess: int
var reply: string(1)
put "Guess the hidden number"
put "It is between 1 and 99 inclusive"
loop
    var count: int := 0
    put "Do you want to play?  Answer y or n" ..
    get reply
    exit when reply = "n"
    % Choose a random number between 1 and 99
    randint (hidden, 1, 99)
    loop
        put "Enter your guess (any number between 1 and 99)" ..
        get guess
        count := count + 1
        if guess < hidden then
            put "You are low"
        elseif guess > hidden then
            put "You are high"
        else
            put "You got it in ", count, " guesses"
            exit
        end if
    end loop
end loop
```



3.6 (5-00) A blank line need not precede or follow computer notation within text (hereafter referred to as embedded computer notation). However, embedded computer notation should be transcribed in Computer Braille Code. If embedded computer notation will not fit entirely on the current line, but will fit on a whole braille line, this notation should begin on a new line. Longer embedded notation should begin on the current line and be divided at a logical place such as following punctuation or a complete word. Runovers should be written at the margin currently in effect. Text which follows embedded computer notation may resume on the current line. The begin Computer Braille Code indicator must begin embedded computer notation except when the first symbol of such notation is the caps lock indicator, the begin emphasis indicator, the begin Nemeth Code indicator, or the begin shape indicator. A termination indicator must always be used within embedded computer notation to signal a return to primary code and/or to cancel the effect of the most recent instance in which the termination indicator applies. (For discussions of the caps lock, emphasis, Nemeth Code, shape, and termination indicators, see Sections 4.0, 10.0, 12.0, 13.0, and 16.0, respectively.)

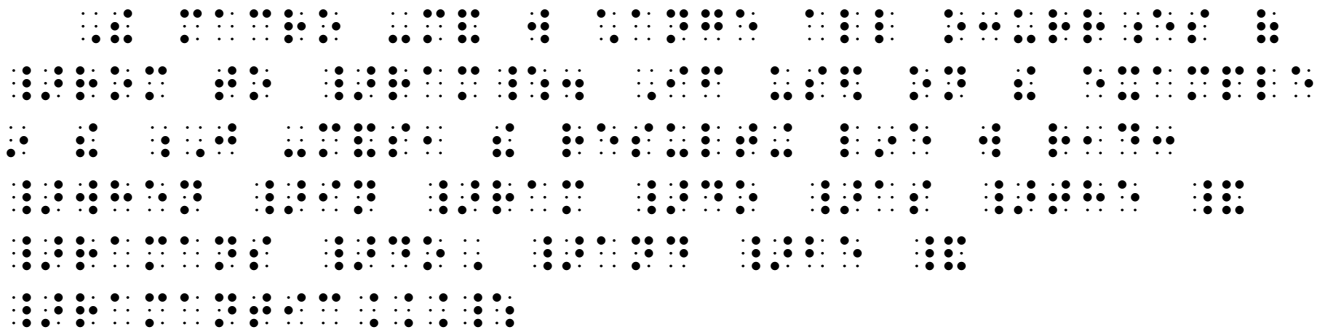
Example 3.6.1: Embedded Computer Notation

To specify all files named ACCOUNTS, regardless of extension, type the following:



Example 3.6.2: Embedded Computer Notation with Runover

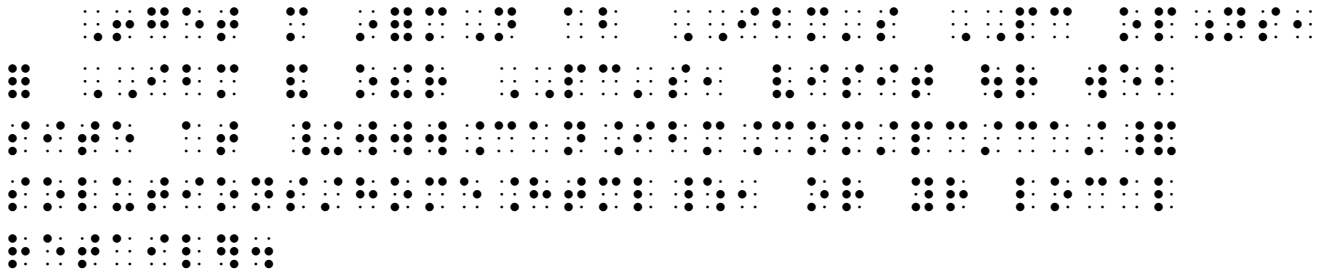
The macro command will change all occurrences of ROM to RAM. If used on the example in the "J" commands, the resulting line will read: WHEN IN RAM DO AS THE RAMANS DO, AND BE RAMANTIC...



Note: The words ROM and RAM are part of a discussion of text editing and are not used as acronyms.

Example 3.6.3: Embedded Computer Notation (Web Site) with Runover (5-00)

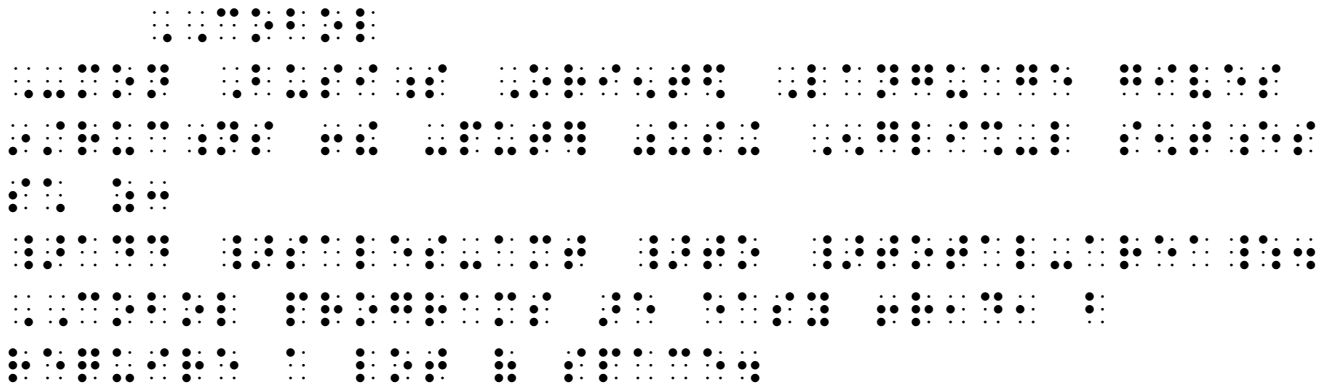
To get more information about IBM's PC options, for IBM and other PCs, visit our web site at www.can.ibm.com/pc/ca/solutions/home.html, or your local retailer.



Example 3.6.4: Embedded Computer Notation on New Line (5-00)

COBOL

Common Business Oriented Language gives instructions to the computer by using English-like sentences such as: ADD SALES-AMT TO TOTAL-AREA. COBOL programs are easy to read, but require a lot of space.

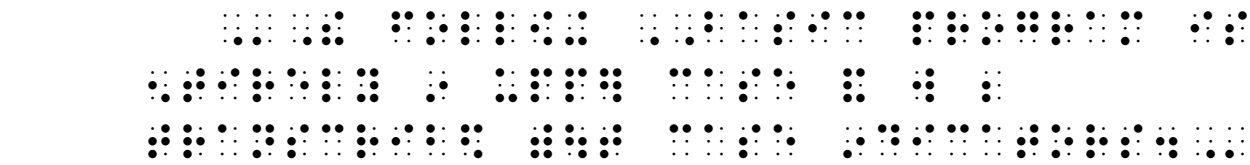


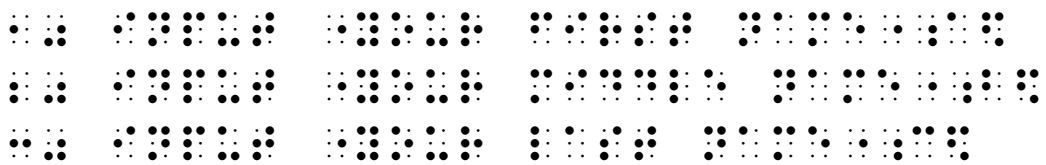
4.0 UPPER AND LOWER CASE DESIGNATION

4.1 Unless otherwise indicated, the Computer Braille Code assumes that letters and words contained in computer notation are lower case. When computer notation is predominantly or exclusively upper case (e.g., a BASIC text), the transcriber may elect to use single-cell letters to indicate upper case and letters preceded by dots 456, the shift indicator, to designate lower case. If computer notation throughout a text is exclusively upper case, a transcriber's note to this effect should appear on the "Transcriber's Notes" page. (Examples 1.1.2, 1.2.1, 1.3.1d, 1.4.1, 1.7.1, 5.1.1, 6.3.1, 8.1.1, 9.1.1, and 14.2.1 would include such a note.) If computer notation throughout a text is not exclusively upper case, the transcriber's note should immediately precede the material to which it applies.

Example 4.1.1: Single BASIC Program Transcribed without Case Indicators

10 INPUT "YOUR FIRST NAME";A$
20 INPUT "YOUR MIDDLE NAME";B$
30 INPUT "YOUR LAST NAME";C$





4.2 Shift Indicator ⠸ (456)

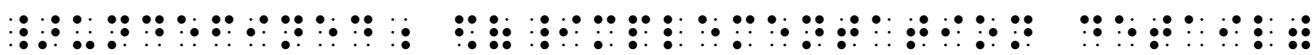
The shift indicator precedes a single upper case letter appearing in embedded computer notation and/or computer notation set off from the body of the text which is otherwise in lower case. Two or more upper case letters occurring in predominantly lower case text—whether these letters appear consecutively or are separated by numbers, punctuation, etc.—must be preceded by the caps lock indicator (see Section 4.3.).

4.3 Caps Lock Indicator ⠨⠠ (456, 345)

The caps lock indicator is used in instances in which embedded computer notation and/or computer notation set off from the body of the text appears in a combination of upper and lower case. The caps lock indicator remains in effect until cancelled by the caps release indicator or the next significant space, whichever comes first. Neither cells occupied by the continuation indicator, spaces following the continuation indicator, nor the space showing indentation of a runover are considered significant spaces.

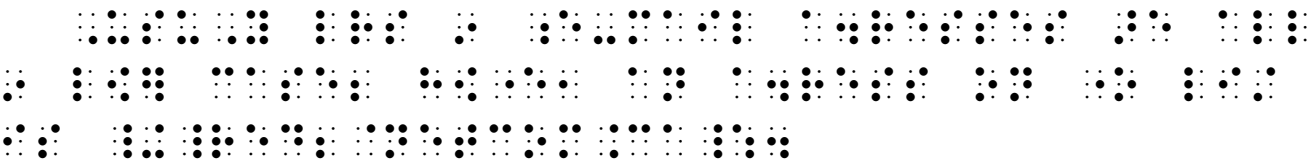
Example 4.3.1: Consecutive and Single Upper Case Letters

UNDEFINED; $(Implementation detail)



Example 4.3.2: Case Designation in E-mail Addresses (5-00)

Usually letters in e-mail addresses are all in lower case; however an address on one list is Red2@netcom.ca.

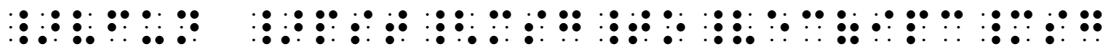


4.4 Caps Release Indicator ⠨⠡ (456, 126)

The caps release indicator cancels the effect of the caps lock indicator.

Example 4.4.1: Caps Lock and Caps Release Indicators

VFUN PSTmsgToVec(ipcMsg



5.0 REPRESENTATION OF CONTROL CHARACTERS

5.1 Control characters, whether in ASCII or other computer codes, are not actually printed by computers and are therefore symbolized in print in a variety of ways. For example, the carriage return function may be represented by "RETURN ," "ENTER ," "CR ," a downward left-curved arrow, a depicted key, or in various other ways. Thus, the braille transcription should follow the print symbology literally, in accordance with the other rules of this code.

Example 5.1.1: Carriage Return Representation

) CREATE/I DSWITCH.CLI ↩



Note: The downward left-curved arrow is used as a carriage return. Therefore, in this example the letters cr are used within the shape indicators.

Example 5.1.2: "Carriage Return" Representation





6.0 CONTINUATION INDICATOR

6.1 Continuation Indicator ⠤⠤⠤ (456, 12346)

The continuation indicator is a line division sign. It is used in instances in which the linear arrangement of the print is significant and the braille line is not long enough to finish the print line. Line division may occur within embedded computer notation as well as within computer notation set off from the body of the text and may generally occur at convenient points (i.e., all available spaces on the braille line need not be filled). If no spaces are present at the point of line division, the continuation indicator should immediately follow the last cell of the braille line (i.e., no spaces should intervene). The continuation indicator may not be used between the cells of a multi-cell sign (e.g., between the first and second cells of the "left brace" symbol). When used to divide countable spaces indicators, the continuation indicator must be preceded by two full cells (123456, 123456). The continuation indicator is not considered a countable space. (See Section 9.0 for a discussion of the countable spaces indicator and its use.)

Example 6.1.1: Continuation Indicator

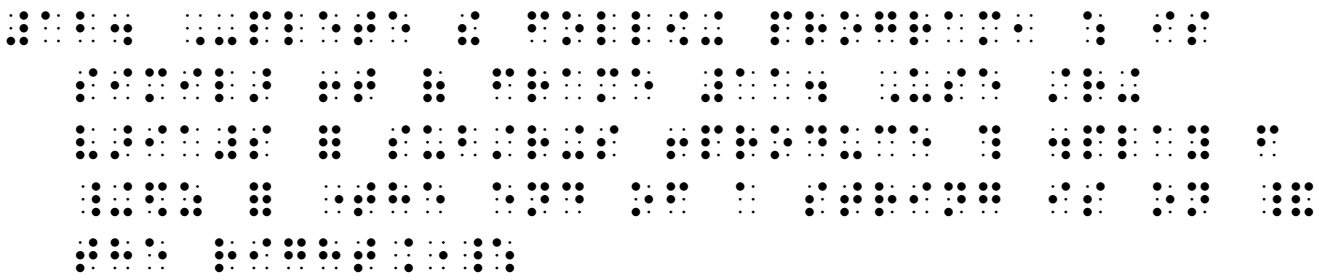
OFUN PSTipcqPut(seid sender,receiver;evType ev;BOOLEAN pseudoInt;

⠤⠤⠤

6.3 A continuation indicator should be used when the entire print line will not fit on the braille line(s), even if the line division occurs following a complete word.

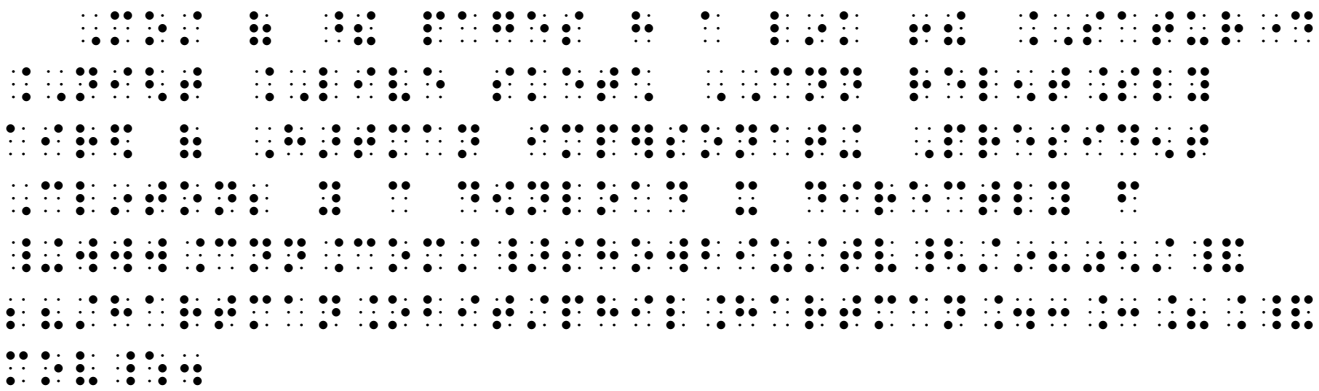
Example 6.3.1: Continuation Indicator within Embedded Computer Notation

12. Complete the following program, which is similar to that of frame 11. Use string variables for substrings to produce this display from $Z = "THE END OF A STRING IS ON THE RIGHT."



Example 6.3.2: Continuation Indicator within Embedded Web Site Address (5-00)

Most of these pages have a link to the *Saturday Night Live* sketch CNN relentlessly aired of Hartman impersonating President Clinton; you can download it directly from **www.cnn.com/SHOWBIZ/TV/9805/28/hartman.obit/phil.hartman.43.3.8.mov**.



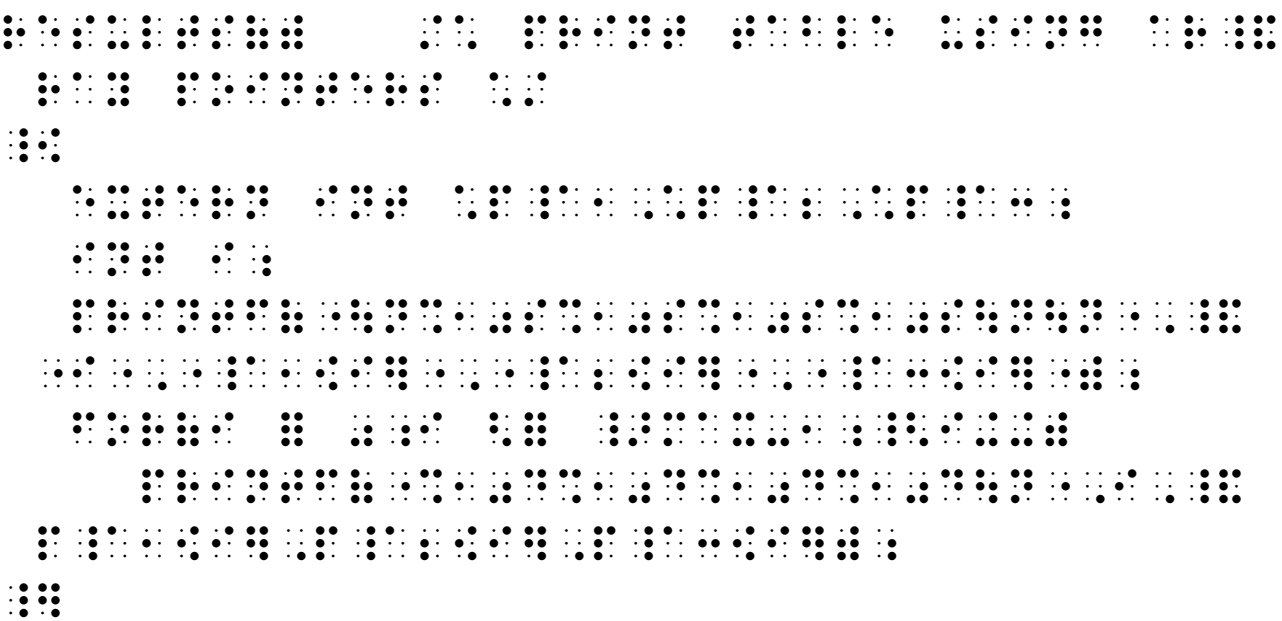
7.0 INDENTED MATERIAL

7.1 When successive levels of indentation occur in print, braille lines should also be indented. Each braille level of indentation should begin two spaces to the right of the previous level, regardless of the print spacing used. Thus, successive levels of braille indentation begin in cells 1, 3, 5, etc. When in doubt, consider spaces at the left as levels of indentation, not countable spaces.

7.2 Runovers should always begin in cell 2, regardless of the current depth of indentation.

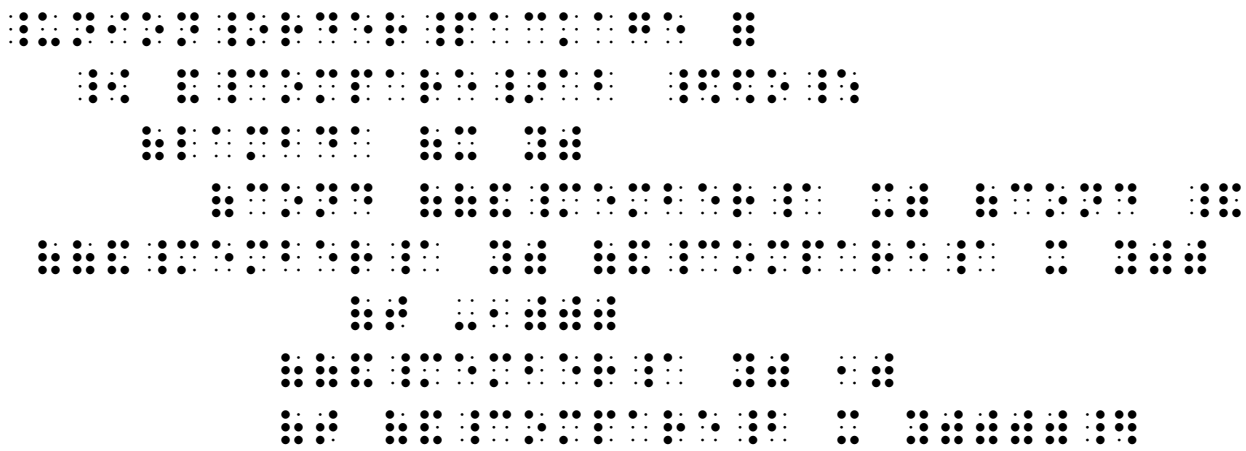
Example 7.2.1: Indented Material

```
results()    /* print table using array pointers */
{
        extern int *pA1,*pA2,*pA3;
        int i;
        printf("\n%10s%10s%10s%10s\n\n","i","A1[i]","A2[i]","A3[i]");
        for(i = 0;i  <=  MAX-1;i++)
                printf("%10d%10d%10d%10d\n",i, pA1[i],pA2[i],pA3[i]);
}
```



Example 7.2.2: Indented Material

```
UnionOrderPackage =
  { &CompareAB →
   (lambda (x y)
        (cond ((&MemberA x) (cond ((&MemberA y) (&CompareA x y))
                                  (t -1)))
              ((&MemberA y) 1)
              (t (&CompareB x y)))))}
```



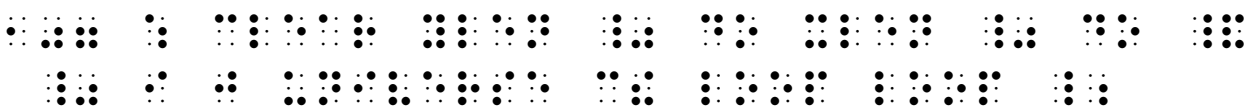
Note: The symbol for the right-pointing arrow was selected from the Nemeth Code and enclosed within the shape indicators.

8.0 ISOLATED LOWER-CELL SIGNS

8.1 When a lower-cell sign is isolated (i.e., preceded and followed by a space or line end), dots 456 must immediately precede this sign. The lower-cell sign may be a single-digit number, quotation mark, apostrophe, comma, hyphen/minus sign, or semicolon. The resulting two-cell symbol signifies a single print character.

Example 8.1.1: Isolated Lower-Cell Signs

```
107 : CLEAR YLEN 0 DO XLEN 0 DO 0 I J UNIVERSE C! LOOP LOOP ;
```



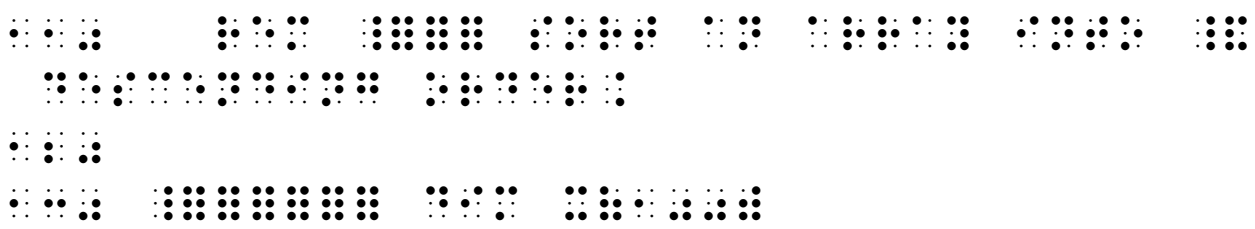
9.0 COUNTABLE SPACES

9.1 Countable Spaces Indicator ⠐⠿⠿⠐ (456), followed immediately by at least two full cells (123456, 123456)

The countable spaces indicator is used when any series of five or more spaces within a print line is significant (e.g., when specific material which is surrounded by spaces must begin on a particular space within a line). These significant spaces, which are important to an accurate understanding of the material being transcribed, are called countable spaces. The first and last spaces of a countable space sequence are transcribed as empty cells (i.e., spaces), with a countable spaces indicator filling the remainder of the spaces to be counted. Each cell of a countable spaces sequence counts as one space, as does each of the spaces surrounding it.

Example 9.1.1: Countable Spaces Indicator

```
110   REM      SORT AN ARRAY INTO DESCENDING ORDER.
120
130         DIM X(100)
```



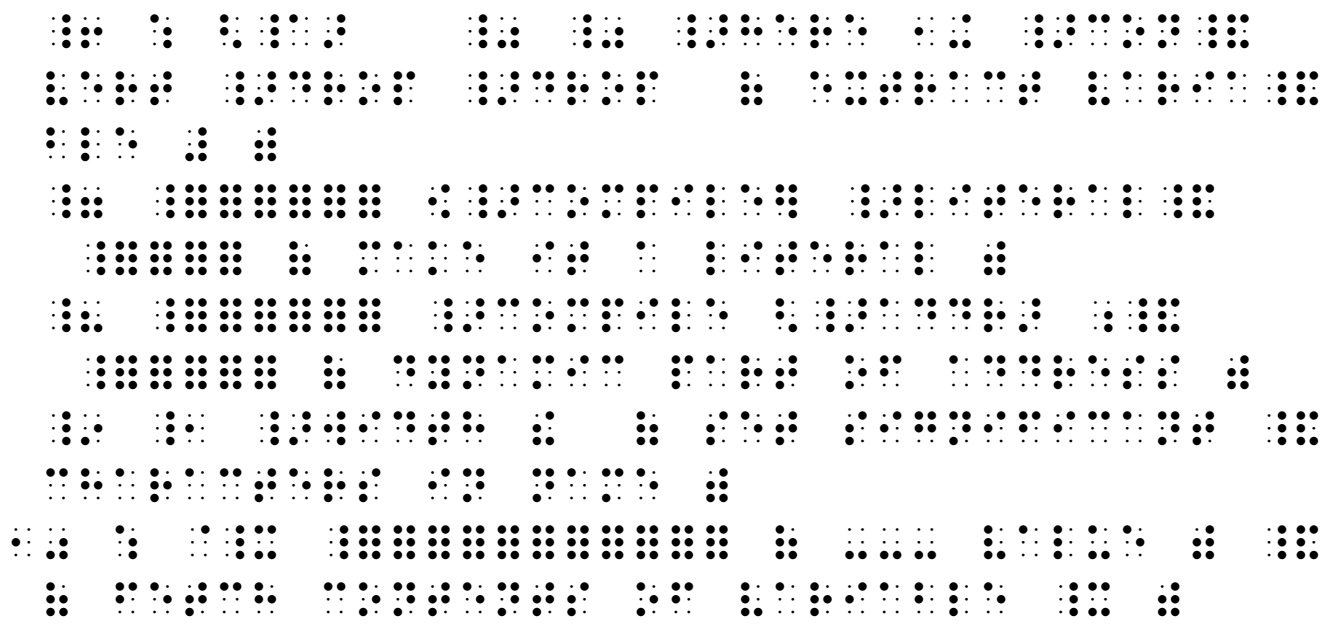
9.2 At least three full cells of the countable spaces indicator must appear on any runover line(s).

Example 9.2.1: Countable Spaces with Continuation Indicator

```
; +--+--+--+
```

Example 9.2.2: Countable Spaces Beginning on Runover Line

```
 6 : <A>   0 0 HERE 1+ CONVERT DROP DROP  ( extract variable # )
 7         [COMPILE] LITERAL       ( make it a literal )
 8         COMPILE <ADDR> ;        ( dynamic part of address )
 9 1 WIDTH !  ( set significant characters in name )
10 : @X              ( --- value ) ( fetch contents of variable X )
```



10.0 EMPHASIS INDICATORS

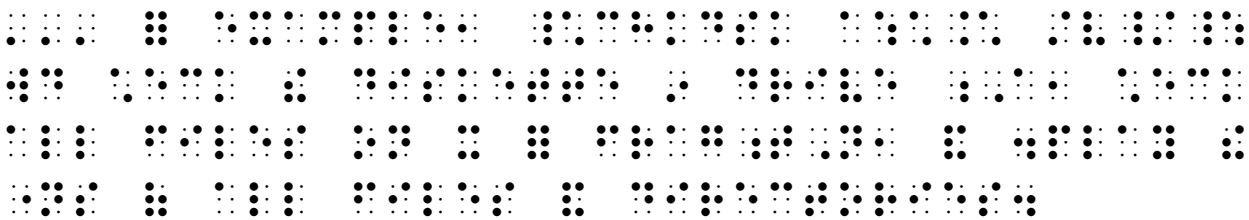
10.1 Begin Emphasis Indicator ⠨⠆ (456, 16)

End Emphasis Indicator ⠨⠔ (456, 34)

Emphasis indicators enclose computer-related material which is emphasized for technical reasons (e.g., to distinguish what a student types from what the computer responds). Print methods of emphasizing this material may include underlining, italicizing, changing type fonts, changing color, etc.

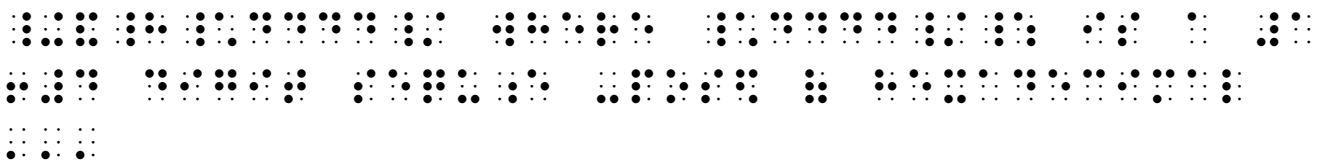
Example 10.1.1: Emphasis Indicators

... for example, *chkdsk a:*.* /v* would check the diskette in drive A, check all files on it for fragmentation, and display the names of all files and directories.



Example 10.1.2: Emphasis Indicators

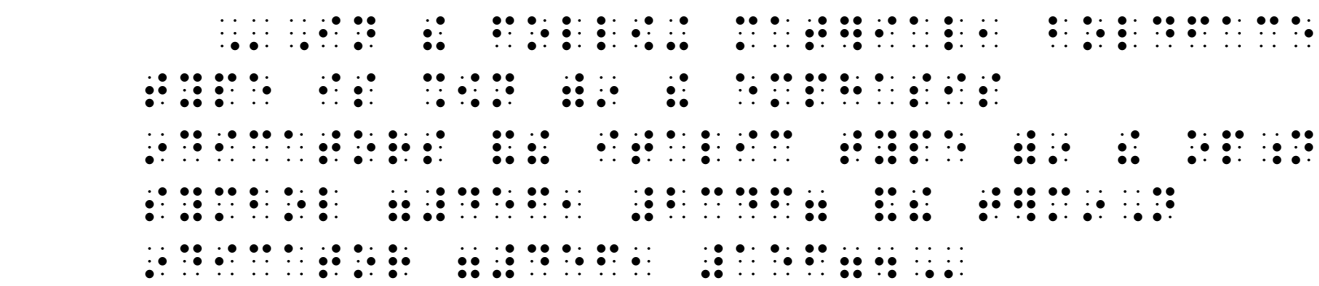
&H*dddd* where *dddd* is a 1 to 4 digit sequence composed of hexadecimal ...

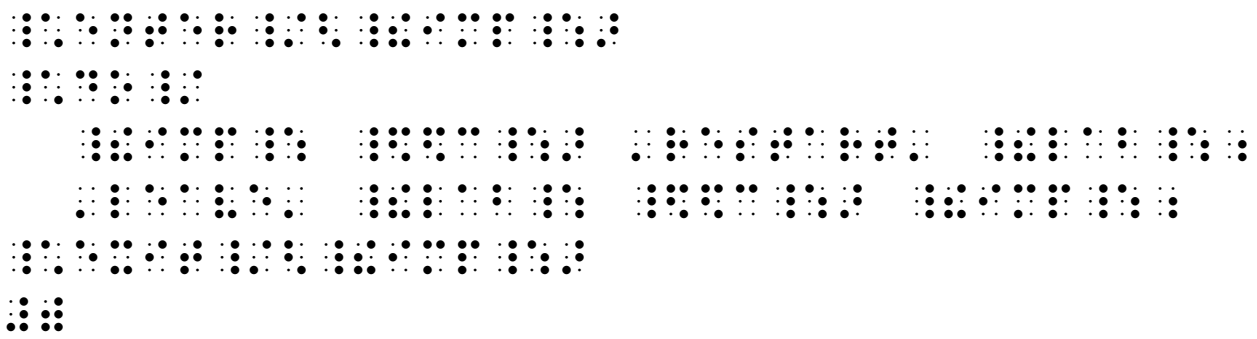


10.2 When more than one type style is used to emphasize technically significant material, a transcriber's option symbol should begin emphasis and a termination indicator should end emphasis in each case. A transcriber's note should explain the use of transcriber's option symbols. (See Section 14.0 for a discussion of transcriber's option symbols and Section 16.0 for a discussion of the termination indicator.)

Example 10.2.1: Emphasis Indicators with More Than One Type Style

enter<*imp*>
do
imp ○> 'restart' *lab*;
'leave' *lab* ○> *imp*;
exit<*imp*>
#)

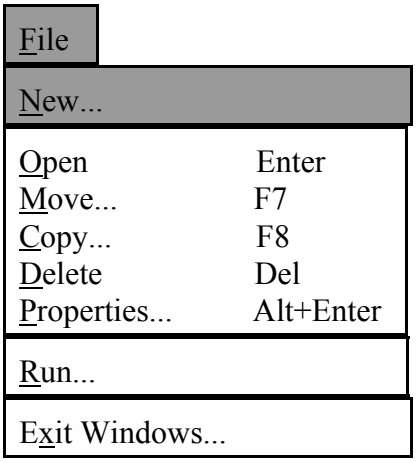


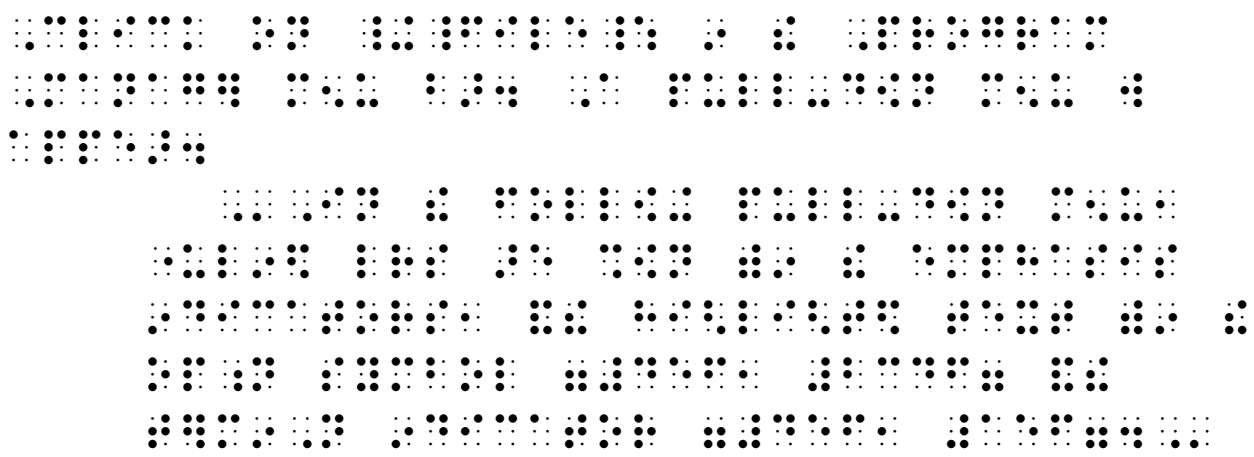


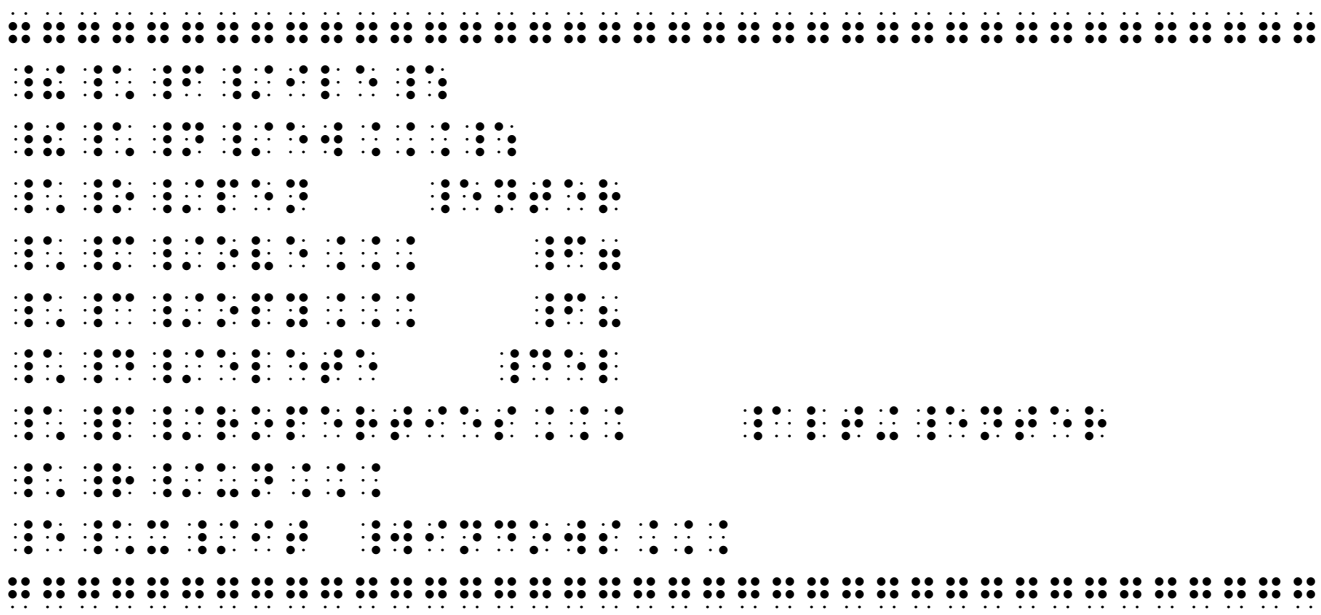
Note: The symbol for the circle was selected from the Nemeth Code and enclosed within the shape indicators.

Example 10.2.2: Emphasis Indicators with More Than One Type Style (5-00)

Click on **File** in the Program Manager menu bar. A pull-down menu will appear.







Note: Since guide dots and empty column fill are not represented in the Computer Braille Code, print column form is not maintained. Where items are present in column 2 they are separated from items in column 1 by 3 spaces.

11.0 COMPUTER BRAILLE CODE INDICATORS

11.1 Begin Computer Braille Code Indicator ⠸⠶⠸ (456, 346)

End Computer Braille Code Indicator ⠸⠶⠼ (456, 156)

Computer Braille Code indicators need not enclose computer notation set off from the body of the text. Computer Braille Code should be entered when textual passages include obvious computer notation (e.g., material to be typed to the computer by the reader in a specific manner).

11.2 If the caps lock indicator, the begin emphasis indicator, the begin Nemeth Code indicator, or the begin shape indicator begins embedded Computer Braille Code, the begin Computer Braille Code indicator need not be used; however, the end Computer Braille Code indicator must always be used to signal a return to regular text. The purpose of Computer Braille Code indicators is to clarify potentially ambiguous situations (e.g., the character preceding s in the filename item.s could be either a period or part of the "less" sign). Frequent switches between Computer Braille Code and the primary code being used should be avoided, particularly within the same braille line. Thus, unless a row or column of a table, a title, or other contiguous material can be transcribed entirely in the primary code, Computer Braille Code should be employed.

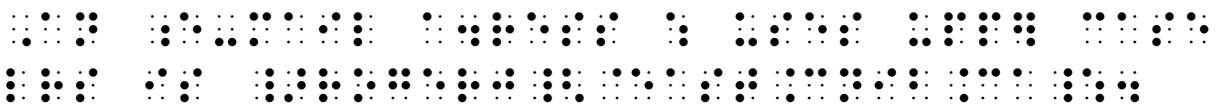
Example 11.2.1: Computer Braille Code Indicators

... the symbols of the form *1, *2, etc.



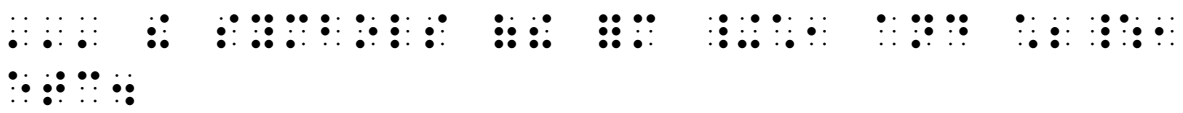
Example 11.2.2: Computer Braille Code Indicators (5-00)

An e-mail address which uses upper case letters is ROGERJ@east.cnib.ca.



Example 11.2.3: Avoiding Frequent Switches between Computer Braille Code and Primary Code

... the symbols of the form *1 and *2, etc.



Example 11.2.4: Use of Computer Braille Code Indicators to Clarify Ambiguities

item.s



Note: This example appeared within the body of the text. Had the computer notation been set off from the body of the text, Computer Braille Code indicators would have been unnecessary.

12.0 NEMETH CODE INDICATORS

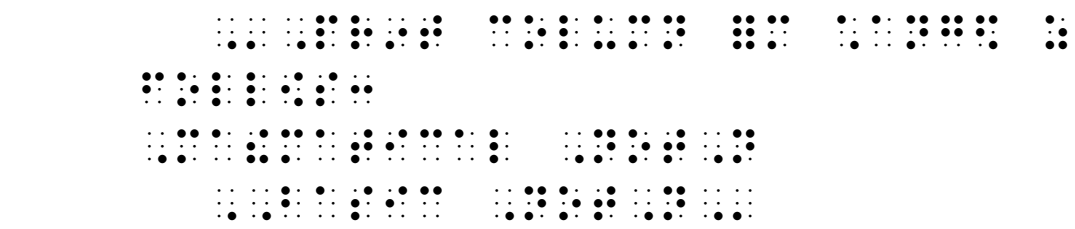
12.1 Begin Nemeth Code Indicator  (456, 146)

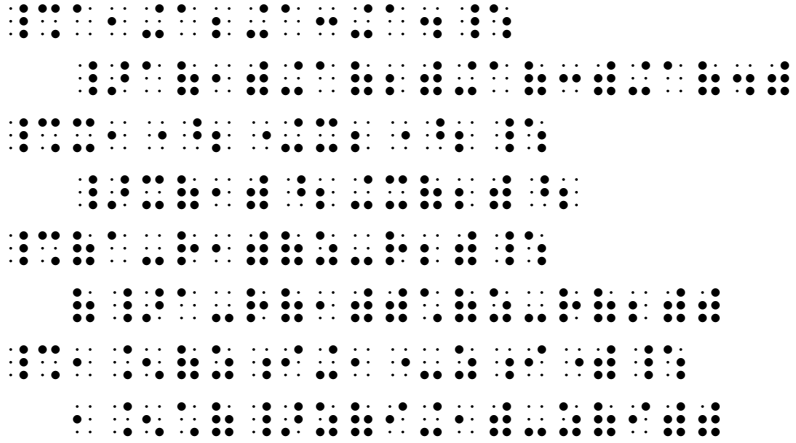
End Nemeth Code Indicator  (456, 156)

Nemeth Code may be necessary when purely mathematical expressions (e.g., formulas, equations, the multiplication symbol when represented by a dot, etc.) are to be transcribed or when no appropriate Computer Braille Code symbol exists (e.g., there is no horizontal fraction indicator in Computer Braille Code). However, Nemeth Code should be resorted to only rarely, and a Nemeth Code expert should be consulted before the transcription is undertaken. If frequent use of Nemeth Code seems necessary, the primary code in which the text is transcribed may need to be changed.

Example 12.1.1: Nemeth Code Indicators

Mathematical Notation	BASIC Notation
$a_1 + a_2 + a_3 + a_4$	A(1)+A(2)+A(3)+A(4)
$x_1^2 + x_2^2$	X(1) ↑ 2+X(2) ↑ 2
$(a-r_1)\ (z-r_2)$	(A-R(1))*(Z-R(2))
$1.5(z_{i+1}-z_i)$	1.5*(Z(l+1)-Z(l))



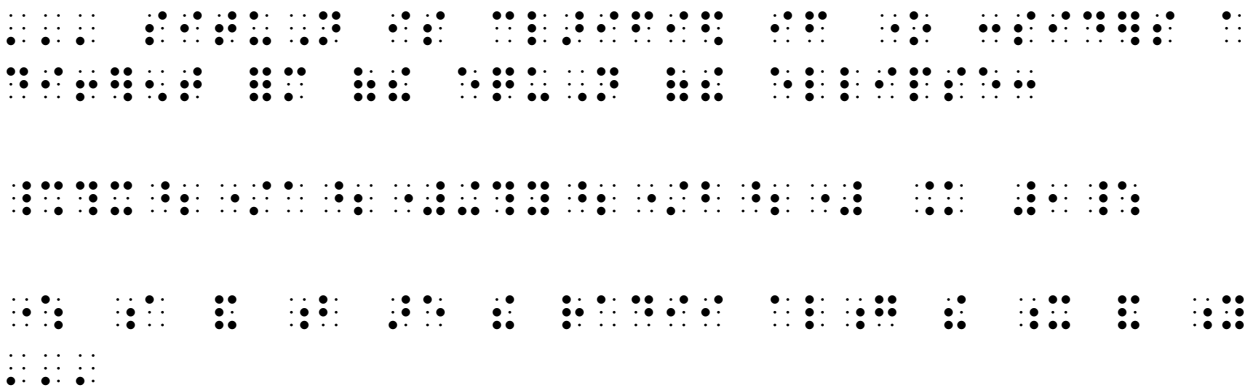


Example 12.1.2: Nemeth Code Indicators with Horizontal Fraction Line

... situation is clarified if one considers a different form of the equation of the ellipse:

$$\frac{x^2}{a^2}+\frac{y^2}{b^2}=1$$

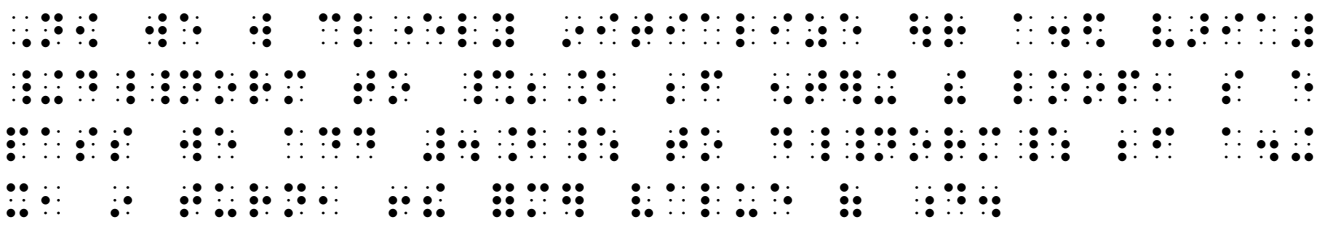
where a and b are the radii along the x and y ...



12.2 Nemeth Code indicators should be written unspaced from the material they enclose. Within Nemeth Code indicators, Nemeth Code notation and rules should be followed. If a long passage includes several Nemeth Code expressions, the begin Nemeth Code indicator should precede the first Nemeth expression. The end Nemeth Code indicator should be written after the last Nemeth expression in the passage has been transcribed. However, Computer Braille Code indicators should enclose computer notation within Nemeth Code expressions. The end Nemeth Code indicator returns the transcription to the baseline of writing.

Example 12.2.1: Nemeth Code Indicators within Computer Braille Code

Now we will cleverly initialize our added variable d_norm to 2β before entering the loop, so every pass we add 4β to d_norm before adding it, in turn, to the former value of d.



13.0 SHAPE INDICATORS

13.1 Begin Shape Indicator ⠸⠶ (456, 1246)

End Shape Indicator ⠸⠶ (456, 156)

Shape indicators are used to enclose a brief series of symbols describing a shape not represented in the Computer Braille Code.

Example 13.1.1: Shape Indicators

S ⊟ R = ⟦ < s, add > | s S - R⟧

⠸⠶ ⠸⠶⠎⠠⠗ ⠸⠶ ⠸⠶ ⠠⠎⠸⠶⠎⠠⠎⠠⠗⠠⠎ ⠠⠎ ⠎⠠ ⠠⠠⠎⠠⠎ ⠎ ⠸⠶ ⠸⠶
⠎ ⠸⠶ ⠸⠶ ⠸⠶⠎⠠⠎⠠⠎⠠⠗⠠⠎

Note: The shape for the square with an interior minus was transcribed as an s followed by a minus and enclosed within the shape indicators. The left and right double-barred brackets were transcribed as lb and rb within the shape indicators.

Example 13.1.2: Shape Indicators (5-00)

To access the scheduler, click on the  icon in the Anti-Virus Toolkit window.

⠠⠠⠎⠠⠎⠠⠎ ⠎⠠ ⠠⠎⠠⠎⠠⠎⠠⠎⠠⠗⠠⠎ ⠠⠎⠠⠗ ⠗⠠ ⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎ ⠠⠎⠠⠎⠠⠎
⠠⠎⠎ ⠠⠎⠠⠎⠠⠎ ⠠⠎⠠⠎⠠⠎ ⠠⠎⠠⠗ ⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎ ⠠⠎⠠⠎
⠠⠎⠠⠎⠠⠎⠠⠎ ⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎
⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎ ⠎⠠ ⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎
⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎ ⠎⠠ ⠎⠠ ⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎⠠⠎

Example 13.2.1: Depicted Key

Hit [ENTER] key.

⠓⠊⠞ ⠠⠑⠝⠞⠑⠗ ⠅⠑⠽

13.3 Transcribers familiar with Nemeth Code may select or devise symbols in accordance with Nemeth Code conventions, thus enhancing consistency among braille codes.

Example 13.3.1: Use of Nemeth Code Symbol of Shape within Shape Indicators

Characters existing only in the 'A' set:

& # % □ @

Equivalent characters from ...

⠠⠠⠝⠉⠃ ⠠⠠⠍⠁⠞⠓ ⠠⠠⠏⠓⠽ ⠯ ⠷ ⠠⠠⠉⠎ ⠠⠠⠎⠽⠍

⠸ ⠳ ⠨ ⠠⠠⠎⠓⠁⠏⠑ ⠐

⠠⠠⠎⠓⠁⠏⠑ ⠠⠠⠊⠝⠙⠊⠉⠁⠞⠕⠗⠎ ⠠⠠⠎⠞⠁⠗⠞ ⠨ ⠠⠠⠑⠝⠙

Note: The symbol for the square was selected from the Nemeth Code and enclosed within the shape indicators.

13.4 A transcriber's note at the beginning of the work or immediately prior to the first use of shape indicators should explain the symbol's meaning (if known), or its shape should be described or drawn. (See Example 13.1.2.)

14.0 TRANSCRIBER'S OPTION SYMBOLS

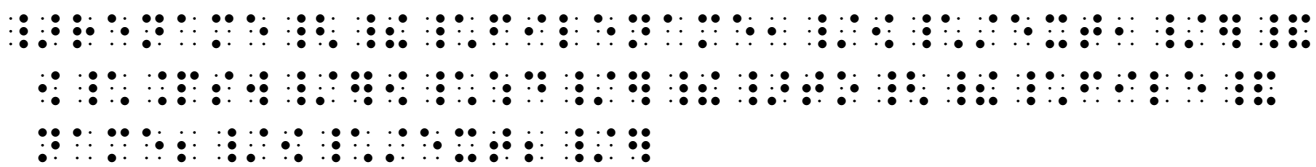
14.1 Primary Transcriber's Option Symbol ⠸⠿⠸ (456, 2346)

Secondary Transcriber's Option Symbol ⠸⠿⠈⠸ (456, 46)

When a character not represented by the Computer Braille Code occurs in a transcription with great frequency, the transcriber may use a transcriber's option symbol to represent that character. The minimum number of spaces required to produce the character is thus reduced from five to two. Transcriber's option symbols should be explained in a transcriber's note.

Example 14.1.1: Transcriber's Option Symbol Representing Non-ASCII Character (~~b~~)

RENAME~~b~~*filename1*[*/ext1*][.*psw*][*:d*]~~b~~TO~~b~~*filename2*[*/ext2*]



14.2 When the print text does not include a character used by the Computer Braille Code but does include a character not represented by the Computer Braille Code, the meaning of an unused Computer Braille Code character may be changed during that transcription (e.g., in an EBCDIC transcription, the unused Computer Braille Code tilde could represent the frequently used EBCDIC "not" sign). A transcriber's note should explain such substitutions.

Example 14.2.1 Transcriber's Substitution (ASCII Tilde Becomes EBCDIC "Not" Sign)

EX = (P ¬= NULL);



15.0 HALF-LINE SHIFTS

15.1 Half-Line Shift Up  (456, 3456)

Half-Line Shift Down  (456, 1456)

End Half-Line Shift  (456, 156)

Half-line shifts up represent superscripts—and in some cases exponentiation—which typically appear within text. If exponentiation within text is indicated by the up arrow, the up arrow symbol (dots 45) should be transcribed. Half-line shifts up terminate either with a termination indicator (see Section 16.0) or at the next space. When the superscript is not followed immediately by a space, the termination indicator must be used.

15.2 Half-line shifts down are used for transcribing subscripts and expressions in which the number base appears below normal line level. Half-line shifts down terminate either with a termination indicator (see Section 16.0) or at the next space. When the subscript is not followed immediately by a space, the termination indicator must be used.

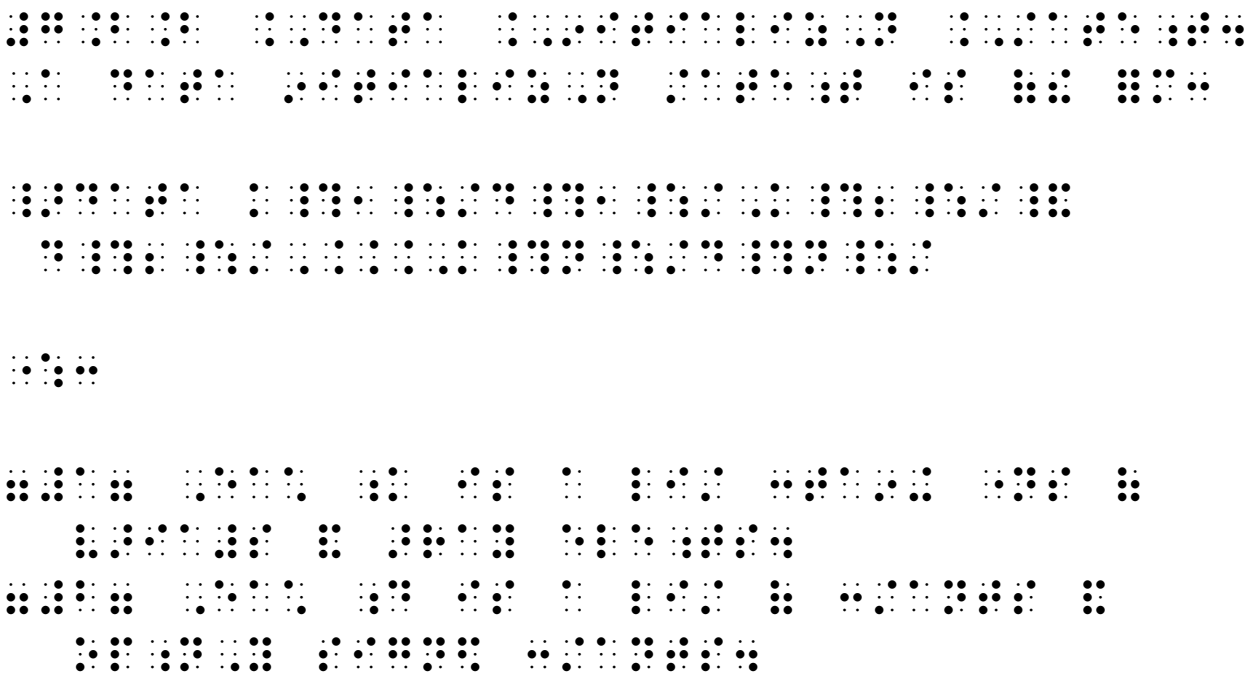
Example 15.2.1: Half-Line Shifts

7.2.2 *Data Initialization Statement.* A data initialization statement is of the form:

DATA $k_1/d_1/,k_2/d_2/,...,k_n/d_n/$

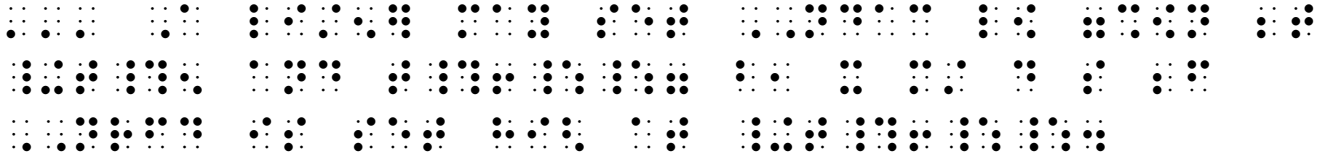
where:

(1) Each k is a list containing names of variables and array elements.
(2) Each d is a list of constants and optionally signed constants.



Example 15.2.2: Half-Line Shift Terminated by a Space

... A listener may set NDAC low (shown between t_5 and t_6) but, it must do so before NRFD is set high at t_6.

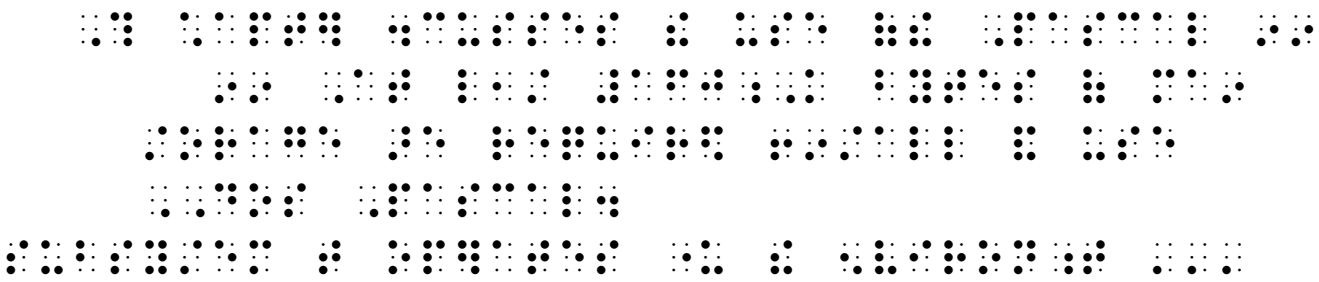


15.3 Reference indicators designating footnotes are transcribed in the primary code being used, not as half-line shifts up.

Example 15.3.1: Reference Indicator (English Braille)

This chapter discusses the use of the Pascal* subsystem that operates under the environment ...

*At least 160K bytes of main storage are required to install and use DOS Pascal.

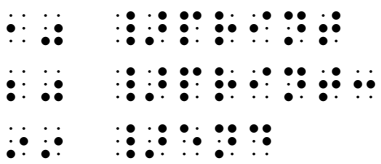


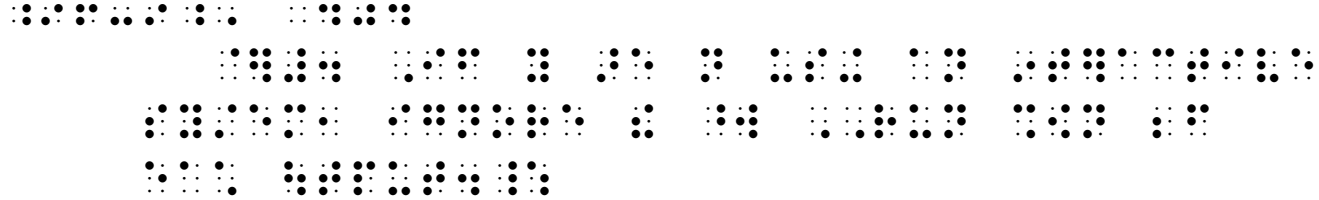
Example 15.3.2: Reference Indicator (Nemeth Code)

10 PRINT
20 PRINT3
99 END

RUN[4]

4. If you are not using an interactive system, ignore the word RUN shown before each output.





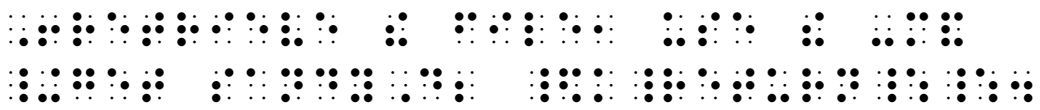
16.0 TERMINATION INDICATOR

16.1 Termination Indicator ⠸⠸ (456, 156)

The termination indicator is context dependent and terminates embedded computer notation, Nemeth Code, a shape indicator, half-line shifts up, and half-line shifts down. If termination indicators must be nested (i.e., if one or more termination indicators intervene before the effect of the first lapses), the first termination indicator cancels the effect of the most recent symbol to which it applies. For example, embedded computer notation including an entering computer code indicator to transcribe a filename and a begin shape indicator to transcribe a carriage return represented as a depicted key would require the use of two termination indicators—one to exit Computer Braille Code, the other to terminate the shape indicator. The first termination indicator would close the shape indicator (the most recent symbol), while the second would exit Computer Braille Code. Punctuation preceding the final termination indicator is transcribed in the code in use at the time. Punctuation following the final termination indicator is transcribed in the primary code being used.

Example 16.1.1: Termination Indicator

To retrieve the file, use the command `get sandy,d2` Return .



COMPUTER BRAILLE CODE SYMBOLS: ASCII ORDER

Note: Upper case letters may be transcribed as shown in the table, preceded by dots 456, or preceded by the caps lock indicator; applicable Computer Braille Code rules determine transcription method. "Reserved" designates currently unassigned symbols which are reserved for future use.

Print	Braille	Dots	Meaning
			space
!		2346	exclamation point
"		5	quotation marks, double quotes
#		3456	number sign, pound sign (weight)
$		1246	dollar sign
%		146	percent sign
&		12346	ampersand
'		3	apostrophe, single quote
(		12356	left parenthesis
)		23456	right parenthesis
*		16	asterisk, multiplication sign

COMPUTER BRAILLE CODE

Print	Braille	Dots	Meaning
+		346	plus sign
,		6	comma
-		36	hyphen, minus sign
.		46	period, decimal point
/		34	slash, division sign
0		356	0
1		2	1
2		23	2
3		25	3
4		256	4
5		26	5
6		235	6
7		2356	7

Print	Braille	Dots	Meaning
8		236	8
9		35	9
:		156	colon
;		56	semicolon
<		126	less than
=		123456	equals
>		345	greater than
?		1456	question mark
@		4	at
A		1	A
B		12	B
C		14	C
D		145	D
E		15	E

COMPUTER BRAILLE CODE

Print	Braille	Dots	Meaning
F		124	F
G		1245	G
H		125	H
I		24	I
J		245	J
K		13	K
L		123	L
M		134	M
N		1345	N
O		135	O
P		1234	P
Q		12345	Q
R		1235	R
S		234	S

Print	Braille	Dots	Meaning
T		2345	T
U		136	U
V		1236	V
W		2456	W
X		1346	X
Y		13456	Y
Z		1356	Z
[		246	left bracket
\		1256	backslash
]		12456	right bracket
↑		45	up arrow/caret
_		456, 456	underscore
`		456, 4	grave accent

Print	Braille	Dots	Meaning
a		1	a
b		12	b
c		14	c
d		145	d
e		15	e
f		124	f
g		1245	g
h		125	h
i		24	i
j		245	j
k		13	k
l		123	l
m		134	m
n		1345	n

Print	Braille	Dots	Meaning
o		135	o
p		1234	p
q		12345	q
r		1235	r
s		234	s
t		2345	t
u		136	u
v		1236	v
w		2456	w
x		1346	x
y		13456	y
z		1356	z
{		456, 246	left brace
|		456, 1256	vertical bar

Print	Braille	Dots	Meaning
}		456, 12456	right brace
~		456, 45	tilde

Additional Computer Braille Code Symbols

Print	Braille	Dots	Meaning
		456, 2346	transcriber's option symbol
		456, 46	transcriber's option symbol
		456, 3456	half-line shift up
		456, 1456	half-line shift down
		456, 156	end half-line shift
		456, 1246	begin shape indicator
		456, 156	end shape indicator
		456, 346	begin Computer Braille Code
		456, 156	end Computer Braille Code

Print	Braille	Dots	Meaning
		456, 146	begin Nemeth Code
		456, 156	end Nemeth Code
		456, 12346	continuation indicator
		456, 12356	reserved
		456, 23456	reserved
		456, 123456	reserved
		456, 16	begin emphasis
		456, 34	end emphasis
		456	shift indicator, isolated lower-cell indicator
		456, 345	caps lock indicator
		456, 126	caps release indicator
		(space) 456, 123456, 123456, ... (space)	countable spaces indicator

COMPUTER BRAILLE CODE SYMBOLS: BRAILLE ORDER

Note: Upper case letters may be transcribed as shown in the table, preceded by dots 456, or preceded by the caps lock indicator; applicable Computer Braille Code rules determine transcription method. "Reserved" designates currently unassigned symbols which are reserved for future use.

Print	**Braille**	**Dots**	**Meaning**
			space
A or a		1	A or a
B or b		12	B or b
C or c		14	C or c
D or d		145	D or d
E or e		15	E or e
F or f		124	F or f
G or g		1245	G or g
H or h		125	H or h
I or i		24	I or i
J or j		245	J or j
K or k		13	K or k

COMPUTER BRAILLE CODE

Print	Braille	Dots	Meaning
L or l		123	L or l
M or m		134	M or m
N or n		1345	N or n
O or o		135	O or o
P or p		1234	P or p
Q or q		12345	Q or q
R or r		1235	R or r
S or s		234	S or s
T or t		2345	T or t
U or u		136	U or u
V or v		1236	V or v
X or x		1346	X or x
Y or y		13456	Y or y
Z or z		1356	Z or z

Print	Braille	Dots	Meaning
&		12346	ampersand
=		123456	equals
(		12356	left parenthesis
!		2346	exclamation point
)		23456	right parenthesis
*		16	asterisk, multiplication sign
<		126	less than
%		146	percent sign
?		1456	question mark
:		156	colon
$		1246	dollar sign
]		12456	right bracket
\		1256	backslash
[		246	left bracket

COMPUTER BRAILLE CODE

Print	Braille	Dots	Meaning
W or w		2456	W or w
1		2	1
2		23	2
3		25	3
4		256	4
5		26	5
6		235	6
7		2356	7
8		236	8
9		35	9
0		356	0
/		34	slash, division sign
+		346	plus sign
#		3456	number sign, pound sign (weight)

Print	Braille	Dots	Meaning
>		345	greater than
'		3	apostrophe, single quote
-		36	hyphen, minus sign
@		4	at
↑		45	up arrow/caret
		456	shift indicator, isolated lower-cell indicator
A		456, 1	A
B		456, 12	B
C		456, 14	C
D		456, 145	D
E		456, 15	E
F		456, 124	F
G		456, 1245	G

COMPUTER BRAILLE CODE

Print	Braille	Dots	Meaning
H		456, 125	H
I		456, 24	I
J		456, 245	J
K		456, 13	K
L		456, 123	L
M		456, 134	M
N		456, 1345	N
O		456, 135	O
P		456, 1234	P
Q		456, 12345	Q
R		456, 1235	R
S		456, 234	S
T		456, 2345	T
U		456, 136	U

Print	Braille	Dots	Meaning
V		456, 1236	V
X		456, 1346	X
Y		456, 13456	Y
Z		456, 1356	Z
		456, 12346	continuation indicator
		456, 123456	reserved
		(space) 456, 123456, 123456, ... (space)	countable spaces indicator
		456, 12356	reserved
		456, 2346	transcriber's option symbol
		456, 23456	reserved
		456, 16	begin emphasis
		456, 126	caps release indicator

Print	Braille	Dots	Meaning
		456, 146	begin Nemeth Code
		456, 1456	half-line shift down
		456, 156	termination indicator
		456, 156	end Computer Braille Code
		456, 156	end Nemeth Code
		456, 156	end shape indicator
		456, 156	end half-line shift
		456, 1246	begin shape indicator
}		456, 12456	right brace
|		456, 1256	vertical bar
{		456, 246	left brace
W		456, 2456	W
		456, 34	end emphasis

Print	Braille	Dots	Meaning
		456, 346	begin Computer Braille Code
		456, 3456	half-line shift up
		456, 345	caps lock indicator
`		456, 4	grave accent
~		456, 45	tilde
_		456, 456	underscore
		456, 46	transcriber's option symbol
.		46	period, decimal point
"		5	quotation marks, double quotes
;		56	semicolon
,		6	comma

INDEX

Abbreviations, 7
Acronyms, 7, 18
Additional CBC symbols, 60-61
Ambiguities, clarification of, 7, 37, 38
Arrow:
    Downward-curve, 21
    Right-pointing, 26
    Up, 7, 47
ASCII, definition, 7
ASCII Order, symbols, 53-61

Blank lines, 11, 17
Braille Order, symbols, 63-71

"Carriage Return", 21
Caps lock, 17, 20, 37
    Transcriber's note, 19
Caps release, 20
Case designation, 7, 19-20, 53, 63
Computer Braille Code Symbols:
    ASCII Order, 53-61
    Braille Order, 63-71
Continuation indicator, 23-24
Contractions, 7, 9
Control characters, 21
Countable spaces, 23, 29-31
    Dividing lines, 23, 30-31
    Indicator, 23, 29-31

Devising symbols, 11, 43, 44, 45
Dividing lines, 13, 23-24, 30-31

E-mail addresses, 6, 20, 37
Embedded notation, 8, 17-18, 51
    Format, 8, 17-18, 51
    Runovers, 17-18, 23
Emphasis indicators, 17, 33-36, 37

Footnote indicators, 49

Format, 11
    Embedded notation, 8, 17-18, 51
    Program lines, 9, 11-16, 19-21, 23, 25-27, 29-31
Fraction line, horizontal, 41
Frequent switches, 37, 38

Half-line shifts, 9, 47-48, 51
    Terminated by space, 9, 48
Horizontal and vertical spacing, 7
Horizontal fraction line, 41

Identifying, 1-6
    Comments explaining, 3-4
    Dot matrix print, 2
    Enclosure symbols, 4
    Horizontal signs of comparison, 5
    Program lines, 1-3
    Symbol not in primary code, 6
    Type styles, 1
    Words out of context, 6
Indented material, 3, 25-26
Indicators:
    Computer Braille Code, 17, 37
    Caps lock, 17, 20, 37
    Caps release, 20
    Case, 19-20
    Continuation, 23-24
    Countable space, 23, 29-31
    Emphasis, 17, 33-36, 37
    Nemeth Code, 17, 37, 39-41, 49, 51
    Reference, 49
    Reserved, 53, 63
    Shape, 5, 17, 43-44, 51
    Shift, case, 19-20
    Shift, half-line, 47-48, 51
    Termination, 17, 43, 47, 51
Isolated lower-cell signs, 27

Keys, depicted, 21, 43-44, 51

Label for program, 13-16
Lower-cell signs, isolated, 27

Nemeth Code indicators, 17, 37, 39-41, 49, 51
Numbers, 7

Primary code, 7, 11, 49, 51
Printable symbols, 7
Program lines, 1-6, 9, 11-12, 13, 14-16, 19-21, 23, 25-27, 29-31, 34, 43, 45, 49
Programs, 1
    Case, 19
    Complete on page, 13
    Dividing lines, 13
    Example, 2, 3, 9, 12, 14-16, 19, 25-26, 29, 30, 31,34, 49
    Format, 2, 3, 9, 11-12, 13, 14-16, 25-26
    Label, 13-16
    Runovers, 11-16, 25, 26, 30, 31
Punctuation, 7, 51

Reference indicators, footnotes, 49
Reserved indicators, 53, 63
Right-pointing arrow symbol, 26
Runovers, 23
    Countable spaces, 30-31
    Embedded, 17-18, 23
    Indented material, 25, 26
    Program lines, 11-16, 25, 26, 30, 31

Set-off material, 11-16, 23, 38
Shape indicators, 5, 17, 43-44, 51
    Transcriber's note, 43, 44
Shift:
    End half-line, 47-48, 51
    Half-line down, 9, 47-48
    Half-line up, 47
    Shift (case) indicator, 19-20
Spaces, countable, 23, 29-31
    Runovers, 23, 30-31
Spaces, significant, 20, 29-31
Spacing, vertical and horizontal, 7
Special symbols page, 11
Subscripts, 9, 47-49
Superimposed symbols, 5
Superscripts, 47, 49
Systematic indentation, 3, 25-26
Symbols:
    ASCII order, 53-61
    Braille order, 63-71
    Devised, 11, 43, 44, 45
    Printable, 7
    Substitutions, 45
    Superimposed, 5

Termination indicator, 17, 43, 47, 51
To, into and by, 7, 9
Transcriber's notes, 11
    Case, 19
    Print format change, 40
    Shape indicators, 34, 43, 44
    Substitutions, 45
    Transcriber's option symbol, 34, 35, 45
    Type fonts, 33-35
Transcriber's option symbols, 34, 35, 45
Type fonts, 33-35

Vertical and horizontal spacing, 7

Web Site addresses, 8, 9, 13, 18, 24